Journal of
Low Power Electronics and Applications

MDPI

Article

Tolerating Permanent Faults in the Input Port of the Network on Chip Router

Hala J. Mohammed [1], Wameedh N. Flayyih [1,*] and Fakhrul Z. Rokhani [2]

[1] Department of Computer Engineering, College of Engineering, University of Baghdad, Baghdad 10071, Iraq; e.hala87@yahoo.com
[2] System-on-Chip Research Center & MyAgeing Research Institute, Universiti Putra Malaysia, Serdang 43400, Malaysia; fzr@upm.edu.my
* Correspondence: wam.nazar@coeng.uobaghdad.edu.iq; Tel.: +964-7705680600

Received: 7 January 2019; Accepted: 22 February 2019; Published: 27 February 2019

Abstract: Deep submicron technologies continue to develop according to Moore's law allowing hundreds of processing elements and memory modules to be integrated on a single chip forming multi/many-processor systems-on-chip (MPSoCs). Network on chip (NoC) arose as an interconnection for this large number of processing modules. However, the aggressive scaling of transistors makes NoC more vulnerable to both permanent and transient faults. Permanent faults persistently affect the circuit functionality from the time of their occurrence. The router represents the heart of the NoC. Thus, this research focuses on tolerating permanent faults in the router's input buffer component, particularly the virtual channel state fields. These fields track packets from the moment they enter the input component until they leave to the next router. The hardware redundancy approach is used to tolerate the faults in these fields due to their crucial role in managing the router operation. A built-in self-test logic is integrated into the input port to periodically detect permanent faults without interrupting router operation. These approaches make the NoC router more reliable than the unprotected NoC router with a maximum of 17% and 16% area and power overheads, respectively. In addition, the hardware redundancy approach preserves the network performance in the presence of a single fault by avoiding the virtual channel closure.

Keywords: NoC; reliability; permanent faults; fault tolerance

1. Introduction

The advancements in the semiconductor industry towards deep-sub micrometer technologies were motivated by Moore's law in the last decades, allowing the integration of billions of transistors on a single chip [1,2]. These chips embed hundreds of memory modules and functional intellectual property (IP) blocks forming multi/many-processor systems-on-chip (MPSoCs) [2,3]. Increasing processing elements numbers in a single chip presents complexity in interconnection architecture making traditional bus-based interconnections unsuitable in MPSoCs [4]. Therefore, network on chip (NoC) interconnection architecture appeared as an alternative on-chip interconnection in MPSoCs due to its high performance, reusability, scalability, and fault tolerance characteristics [1,4,5]. Figure 1 shows the MPSoCs scheme interconnected by NoC.

NoC [6–9] interconnection consists of a group of shared router nodes joined by shared channels or links, making this structure more efficient than buses. The connections between the routers and their attached processing elements (PEs) are constructed by network interfaces (NIs) [10]. The functionality of an NoC is defined by its topology, routing algorithm, flow control, and switching technique. Switching techniques are used to determine data format to traverse its path, such as the wormhole packet switching technique, which is the most widely adopted technique. Wormhole switching divides

the packet into smaller fixed size parts called flits (flow control information units) and traverses them in a pipelined scheme through the network [9,11,12].

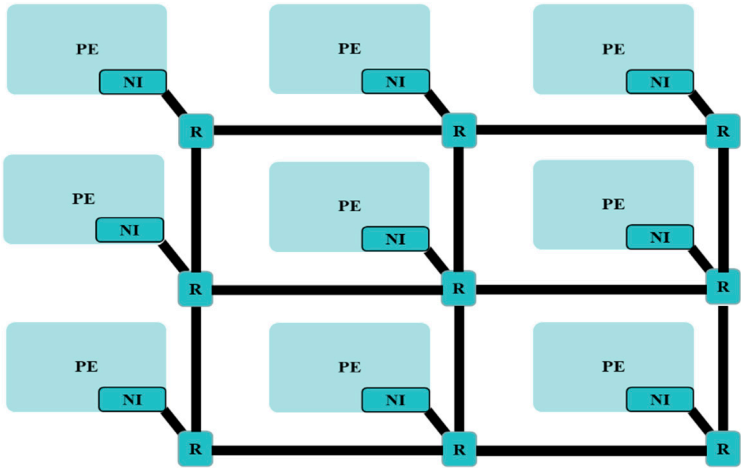

Figure 1. The multi/many-processor systems-on-chip (MPSoCs) scheme interconnected using network on chip (NoC).

By going deeper into the sub micrometer technology and the continuous device downscaling, the on-chip interconnects and routers became more susceptible to faults. There are two different kinds of NoC potential reliability challenges, soft (transient) and hard (permanent) faults. Soft errors, which are unpredictable errors that occur for a short time and are caused by different sources, such as energetic particle strikes [13], electrical noise [14], or process variation [15], may affect the NoC by misrouting packets or making them invalid. Hard faults, which are permanent, persistently affect the circuit functionality from the time of their initiation and are caused by many sources, such as electromigration [16], and time-dependent dielectric breakdown [17].

Many researchers proposed fault tolerance techniques to improve NoC reliability, which is the network probability of delivering messages correctly in a certain time [9]. The faults may occur in some or all components of NoC, namely links and routers. Fault tolerance techniques are achieved mainly by redundancy, which can be time redundancy, such as data retransmission with the selfsame component, information redundancy by adding code for error correction, or spatial redundancy achieved by redundant elements [18]. The soft errors can be countered mainly by error control coding whereas the hard faults can be countered by spare module/gate for replacements, faulty part isolation, or fault tolerant routing [10].

The main part of NoC is the router, which carries out the essential task of steering and coordinating the data flow. It consists of three major parts: input buffers element, routing components, and crossbar component. The flits are stored in the input buffer component and routed by routing components, which are the route computation component, virtual channel allocation component, and switch allocation component. The crossbar component is used to transfer flit to destination node physically [10]. Some works, such as [19–23], addressed the permanent faults in the router by addressing the different components.

The proposed scheme tolerates permanent (hard) faults in very important fields of the input buffer (port) component called the virtual channel state fields, which they track the flits the moment the buffer is reserved until it is released. To the best of our knowledge, these fields have not been considered yet. The proposed scheme uses the hardware redundancy technique for tackling permanent faults in these fields. Furthermore, a built-in-self-test circuitry is integrated into the input port to detect the faults and configure the hardware redundancy circuit accordingly.

The rest of the paper is structured as follows: Section 2 presents related works addressing permanent faults in the NoC router. Section 3 describes the architecture of the baseline 4-stage NoC

router. Section 4 presents the proposed design that tolerates permanent faults in the input buffer component of the NoC router. Section 5 discusses the results of the proposed design in terms of delay, area overhead, power consumption, reliability, and network performance. Section 6 concludes this paper.

2. Related Works

In this section, we review earlier works on fault-tolerant NoC router architectures that focus on tolerating permanent faults in the NoC router pipeline.

N-modular redundancy (NMR) approaches were adopted in the BulletProof router [19] to provide fault tolerance from permanent faults. However, NMR techniques significantly increase the area and power consumption because they require redundant reproductions of hardware. Vicis methodology [20] was suggested to tolerate permanent faults using inherent redundancy at two levels, network and router. At the network level, Vicis approach employs input port swapping supported by an adaptive routing algorithm to tolerate permanent faults. At the router level, it tolerates faults in the crossbar of the router using a bypass bus and in the datapath of the router using error-correcting codes (ECC). Although Vicis router provides better fault tolerance with lesser area overhead than BulletProof [19], it causes a reduction in performance with increasing faults. Furthermore, it can sustain at most one permanent fault at the buffer component.

In [21], the Row-Column (RoCo) Decoupled router was proposed, which divides the router into individual row and column modules. The division process was enabled by using decoupled parallel arbiters and smaller crossbars for row and column connections. Because the row and column modules are independent in their work, a permanent fault in one module does not affect the other module, and the router operates correctly with the fault-free module. The RoCo router tolerates permanent faults in the buffers via a bypass path, in the routing computation stage by using a double route computation in the next router, and in the switch allocation stage by sharing arbiters from virtual channel allocators. Fault tolerance property is not applied for the virtual channel allocation and the crossbar stages. This decomposition strategy is energy-efficient and reliable. However, its recovery designs for faulty modules cause latency penalties and performance degradations.

In [22], the researchers suggested a permanent fault tolerant router (PFTR) and further extended in the Shield router [24], which has the ability to tolerate multiple permanent faults in the pipeline stages. Its reliability is accomplished by adding minimal extra circuitry and exploiting temporal parallelism to individual stages of the baseline router pipeline. The PFTR tolerates permanent faults for the routing computation (RC) stage by using a redundant RC component for each input port, for the virtual channel allocation (VA) stage by using resource sharing, for the switch allocation (SA) stage via bypass path for each arbiter, and for the crossbar (XB) stage by providing two paths for each output port of the crossbar implemented by adding smaller sized decoders of the crossbar. The PFTR module provides better reliability than a baseline router and other fault-tolerant routers. However, the technique of using the idle time of existing fault-free resources degrades performance under high traffic. Furthermore, the PFTR scheme cannot tolerate faults in the input port.

A reliable NoC router based on a 2-stage generic router was presented in [23]. It employs five different fault tolerance techniques for all components in the baseline router. It tolerates permanent faults in the input buffers using error correction code and virtual channel closing technique. It used a double routing technique for the RC component, default winner technique for the VA component, runtime arbiter selection technique for the SA component, and double bypass bus technique for the XB component. The suggested router enhances reliability with lower hardware overhead than earlier reliable router schemes. However, the fault tolerance using ECC detection and virtual channel closing strategy gives an extra cycle latency before transferring flits to the crossbar stage.

Previous approaches mainly focused on tolerating permanent faults in the Routing Computation (RC) component, Virtual channel Allocation (VA) component, Switch Allocation (SA) component, and crossbar (XB) component. In contrast, limited attention was given to permanent faults in the input

buffer component. As a result, providing fault tolerance strategies to this component increases the reliability of the entire router. The fault tolerance technique used in the input buffer unit of our design tolerates faults in the input virtual channel state fields, which play a main role to control the flow of messages in the NoC router.

3. Baseline 4-Stage NoC Router

The baseline router composes of 4-stages pipeline (RC, VA, SA, and switch traversal ST) each one takes one clock cycle to be executed. In this section, the background of a generic 4-stage NoC router is illustrated.

3.1. Router Architecture

Figure 2 shows the architecture of a baseline 4-stage NoC router [9]. The router has five input ports and five output ports, each input port has 1: V demultiplexer, V:1 multiplexer, and V virtual channels (VC's). The control components of the router are the routing computation (RC) component, the virtual channel allocation (VA) component, the switch allocation (SA) component, and the crossbar (XB) which connects the input ports to the output ports of the router [23]. Since the baseline router uses wormhole packet switching technique, the packet is divided into three types of smaller fixed size flits namely a head flit which allocates router resources to the packet, a body flit which contains the packet payload, and a tail flit which de-allocates router resources from the particular packet [22]. Each packet starts with a head flit followed by zero or more body flits and a tail flit. In single flit packets, the head flit is also marked as a tail flit. Flit types are shown in Figure 3.

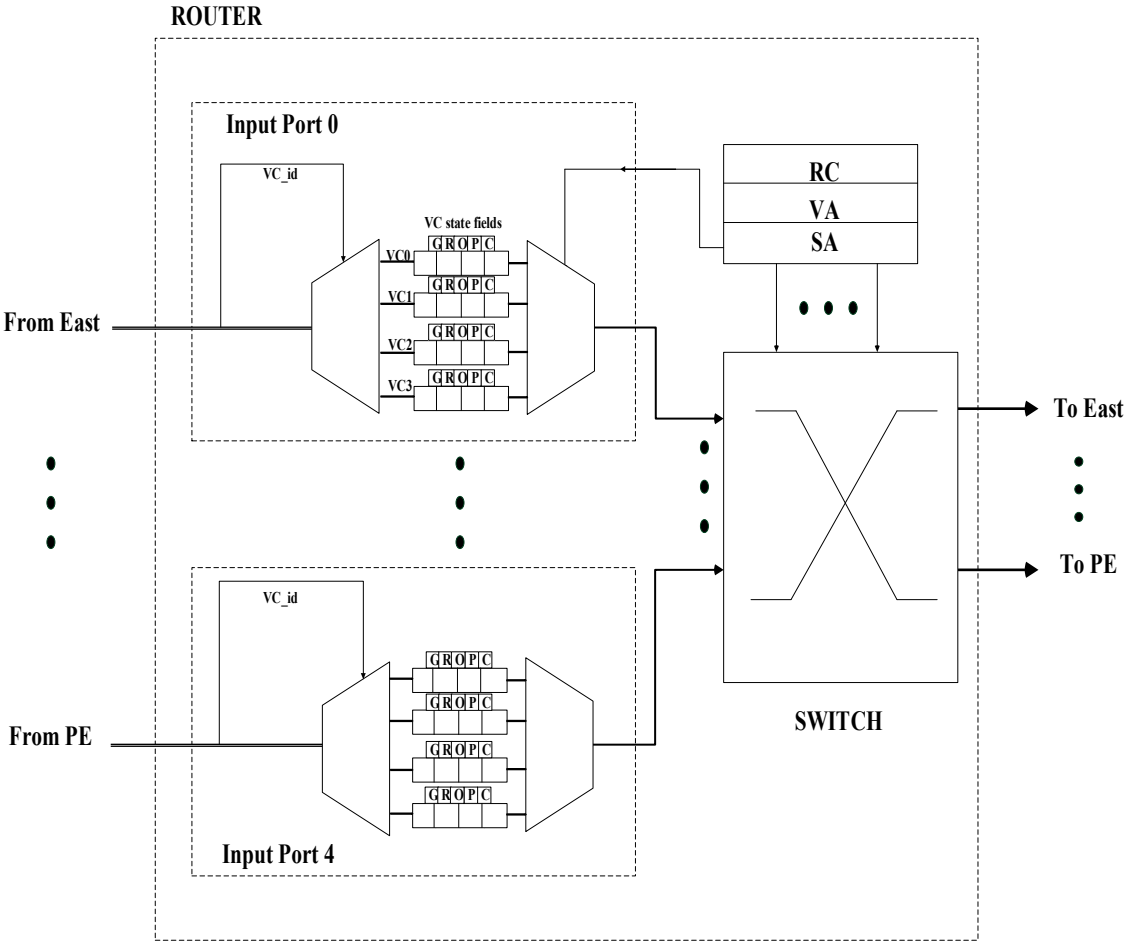

Figure 2. The architecture of a baseline 4-stage NoC router.

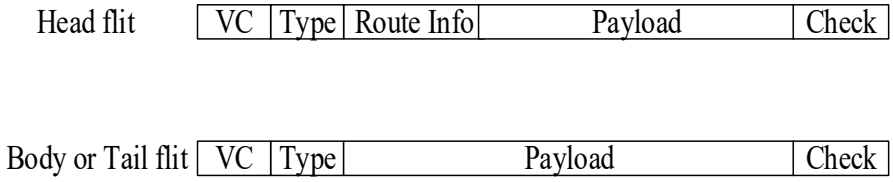

Figure 3. Flit types.

3.2. Router Pipeline

Figure 4 shows the pipeline of the baseline 4-stage NoC router. A packet is processed through the pipeline of the NoC router to be transferred to the next router or to the desired processing element (PE) of that router. The RC and VA stages are called (once per packet) because they perform computation for head flit of each packet. On the other hand, the SA and XB stages are called (once per flit) because they perform computation for all flits (head, body, and tail) of the packet [7]. During the processing of flits in each router component, the virtual channel state fields (G, R, O, P, and C) are changed according to that component as shown in Figure 4.

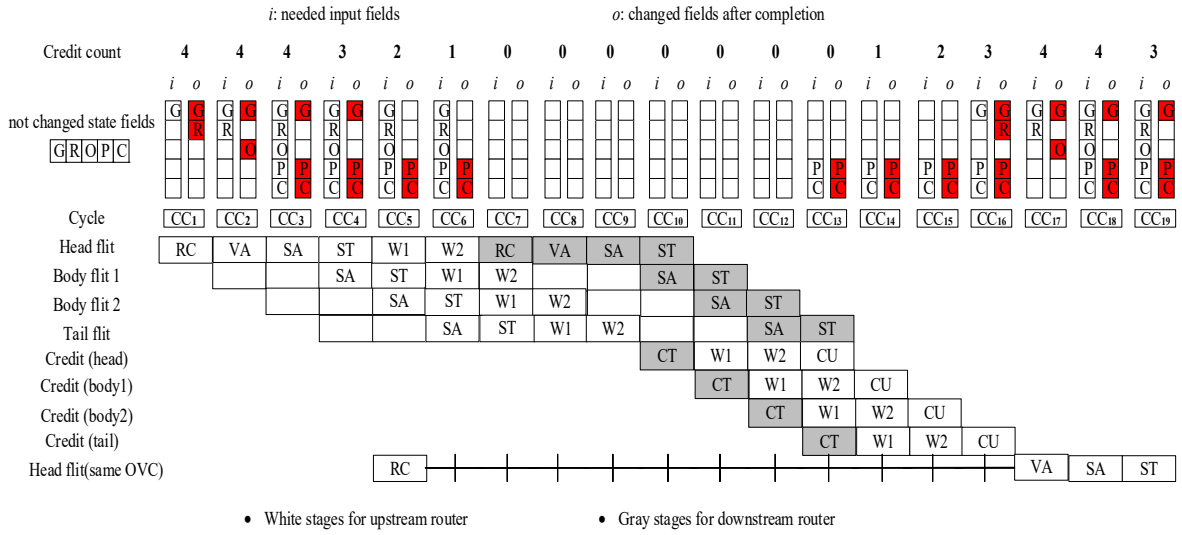

Figure 4. The pipeline of the baseline 4-stage NoC router, assuming two cycles link traversal (W1, W2).

3.3. Architecture of Router Components

3.3.1. Input Buffer Component (Input Port Component)

In a baseline router, each input port is decomposed into groups of flit buffers. These buffers are organized as a compound of fixed size queues as shown in Figure 2. Each queue is named a virtual channel, and all virtual channels for the same input port share a physical channel of that port [23]. Therefore, each new arriving flit is stored in a specific VC buffer selected by VC identifier of that flit. All flits of the same packet are stored in the same virtual channel. Each input virtual channel is tracked by five virtual channel state vectors or fields called Global state (G), Route (R), Output VC (O), Pointers (P) and Credit count (C) fields [9]. The G field represents the status of the current virtual channel and takes one of the following states, either idle (I), routing (R), waiting for an output VC (V), active (A), or waiting for credits (C). The R field stores the output port for the packet and is set after the routing computation stage is completed. The O field holds the output VC of port R assigned to the current input VC, which is the result of the virtual channel allocation stage. The P field stores the head flit and tail flit pointers that point to the buffers in the input VC as shown in Figure 5. The C field indicates the credits count of available in downstream flit buffers for the output virtual channel O on output port R.

Table 1 displays the function of each state field and a summary of the possible effects of faulty fields, which shows the importance of these fields to the flow of messages in the NoC router.

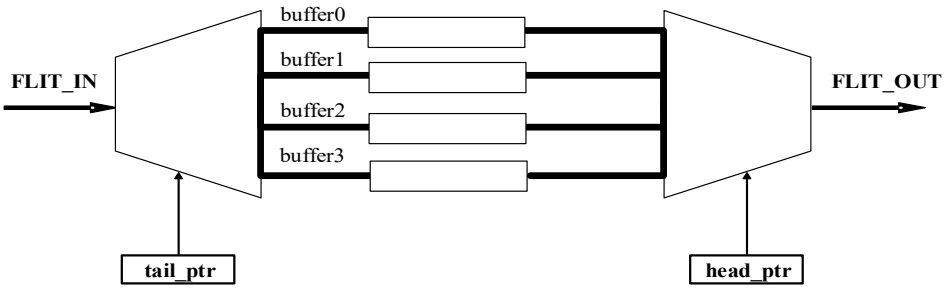

Figure 5. The input virtual channel (VC).

Table 1. Function of the input virtual channel state fields and the effect of faults.

State Field	Function	Faults Effect
G (Global state)	Represents the status of the current virtual channel: idle (I), routing (R), waiting for an output VC (V), active (A), or waiting for credits (C).	The state of VC is changed. Many effects can occur but in general, the pipelined operation will not proceed correctly or may stop.
R (Route)	Stores the output port for the packet and is set after the routing computation stage is completed.	The packet is misrouted.
O (Output VC)	Holds the output VC of port R assigned to the current input VC which is the result of the virtual channel allocation stage.	The wrong O leads to wrong output VC. Packet(s) may be lost.
P (Pointers)	Stores the head flit and tail flit pointers that point to the buffers in the input VC.	The faulted P leads to pointing on the wrong location of buffers. Flits or even packet can be lost, and VC may stay unreleased.
C (Credit count)	Indicates the credits count of available downstream flit buffers for output virtual channel O on output port R.	If C is wrong, then the number of free buffers in output VC is wrong. Flits may be forwarded to a full buffer or flits may stay while the buffer is available.

3.3.2. Routing Computation Logic Component (RC)

In a baseline router, all input ports have one shared routing computation (RC) component, which computes the output port that the packet should be headed to. This decision is taken according to the destination information that exists in the head flit of that packet and on the routing algorithm used. The output port number is stored in the Route (R) field of the virtual channel status fields. The RC component operates only on the head flit of a packet [22].

3.3.3. Virtual Channel Allocation Component (VA)

The virtual channel allocation (VA) component is responsible for allocating the available virtual channel in the downstream router to the arriving packet. This component operates on head flit only and stores its result in the (O) field [22].

3.3.4. Switch Allocation Component (SA)

The switch allocation (SA) component determines which input VC of the input port can transfer its flits through the crossbar component to the output port in the following cycle [22]. It operates on all flits and updates the (C) field when successful.

3.3.5. Switch Traversal Component (Crossbar XB) or (ST)

The crossbar (XB) component connects the input ports with the output ports of the same router. It is controlled by the control signals generated by the switch allocator (SA) [22].

3.4. Alternative Router Pipelines

To reduce the number of cycles in the pipeline, the stages may work in parallel in a single cycle with speculation technique. A pipeline with 3-stages is designed by speculatively performing SA in parallel with VA. This can be further enhanced to a 2-stage pipeline where the flit is sent to the ST stage while the VA and SA are being executed at the same cycle. One stage pipeline is designed by combining the 3-stages are speculation (VA, SA, and ST) with the lookahead RC property, where the next output port is computed in the upstream router and sent with the head flit. These alternatives do not affect the VC state fields, thus, keeping their role crucial in all cases and the proposed protection solution can be incorporated in these pipelines as well. As a result, this work will consider the 4-stage router.

4. Proposed NoC Router

In this section, the proposed router is illustrated, where the focus is on tolerating permanent faults in the state fields of the input buffer component to increase its reliability. Other components or stages were tackled from faults to increase reliability by many researchers [20–23] and can be adopted to provide complete protection. The proposed buffer component employs the hardware redundancy technique to tolerate permanent faults in the input virtual channel state fields (G, R, O, P, and C). In addition to this hardware redundancy technique, the proposed buffer component provides a VC-closing technique and a detection mechanism. The latter is responsible for the detection of faults and controls the hardware redundancy and VC-closing accordingly.

4.1. Detection Mechanism

The proposed detection circuitry represents a built-in-self-test (BIST) mechanism that detects faults, identifies their location, and accordingly provides a decision applied for the virtual channel state fields and for the spare registers. The BIST mechanism aims to detect stuck at 0 and stuck at 1 faults which are the most widely used fault models, then writes the result in the status registers; a 4-bit register State1 for (G, R, O, and C) fields and a 2-bit register State2 for P field (head and tail pointers). Figure 6 shows the operation flow chart of the detection circuitry. The stuck at 1 fault test is first done by applying all zeros value to the testing fields and then checking the output of these fields at the next clock cycle. If the output is all zeros it indicates that there is no stuck at 1 fault found, otherwise the found fault is marked in the corresponding bits of status registers. Stuck at 0 fault test is similarly carried out by applying all ones and looking for faults represented by a zero at the output at the next clock cycle. The same registers are updated according to the test result as well. This mechanism generates the main outputs (test, State1, State2, and vc_closing_out) to be used by the proposed hardware redundancy that will be discussed in the next subsection. The first output (test signal) goes high to start the BIST process when (reset = 1) which means the router operation has started or (G = 3 and vc_release = 1) which means that the packet has left the VC. Otherwise, the test signal goes low, and the normal router operation is resumed. State1 and State2 registers identify the faulted input state field in the proposed hardware redundancy technique. The vc_closing_out signal goes high to close the input virtual channel when it the number of detected faults exceeds the maximum number of tolerated faults in the proposed fault tolerance technique.

The BIST mechanism requires three clock cycles to be completed. In the first cycle, all input VC state fields are set to zero but their outputs will be visible in the next cycle. In the second cycle, the state fields outputs are checked for any stuck-at-1 faults, and their inputs are set to one. In the last cycle, the new state fields outputs are checked for stuck-at-0 faults, and their inputs are set to zero to prepare them for the normal operation in the next cycle. Each input VC has an idle time of ten clock cycles between its release until its next reservation as shown in Figure 4. At this idle time the downstream router waits for credits indicating that the upstream router input VC is free. Therefore, the BIST exploits this idle time to perform a periodic fault detection to avoid interrupting the router operation. For a single stage pipeline with one cycle link traversal, the idle time of each input VC

is four clock cycles (from 7 to 10) between its release until its next reservation as shown in Figure 7, which is still enough to accommodate the proposed BIST.

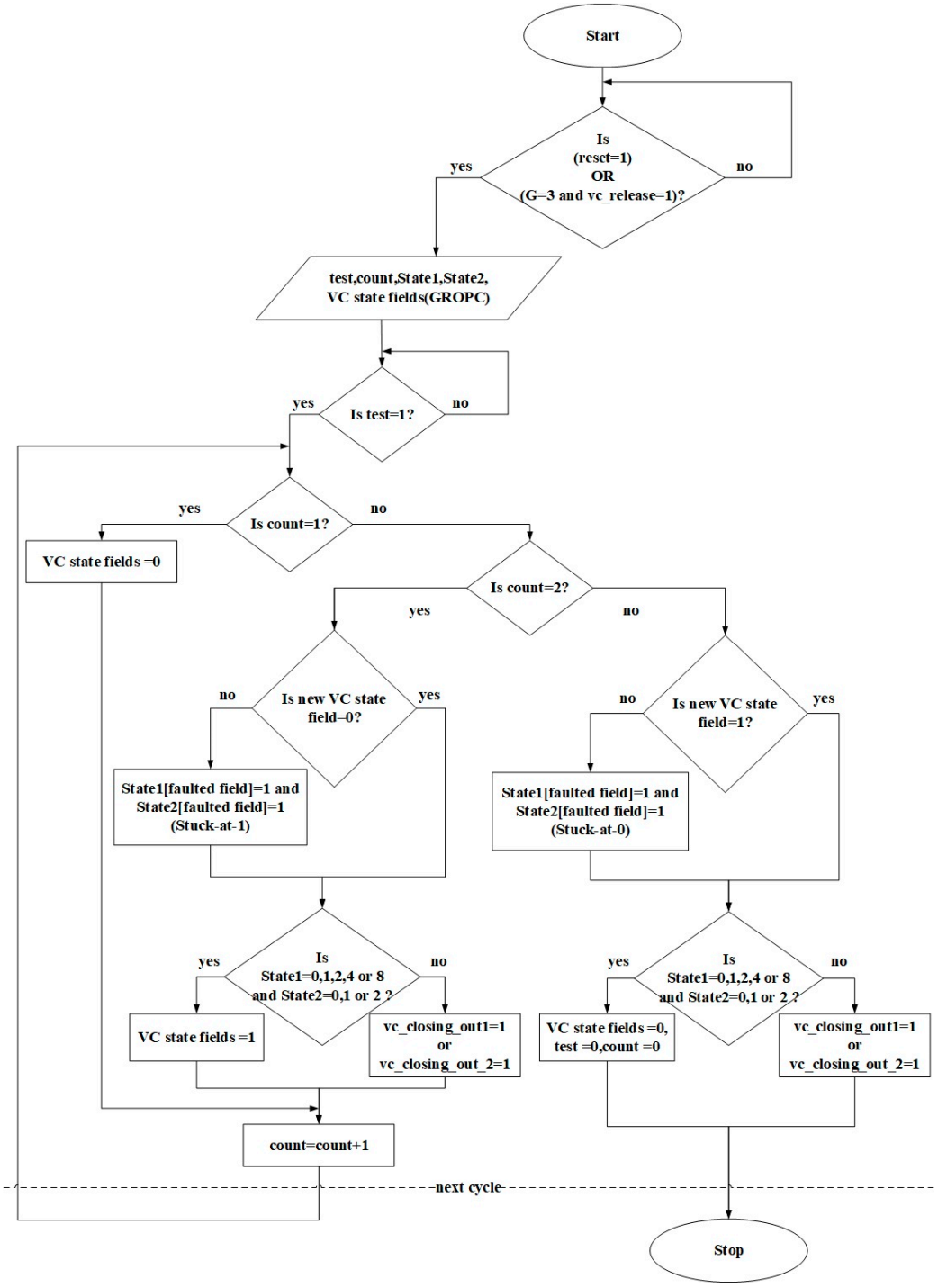

Figure 6. The proposed detection operation (built-in-self-test (BIST)).

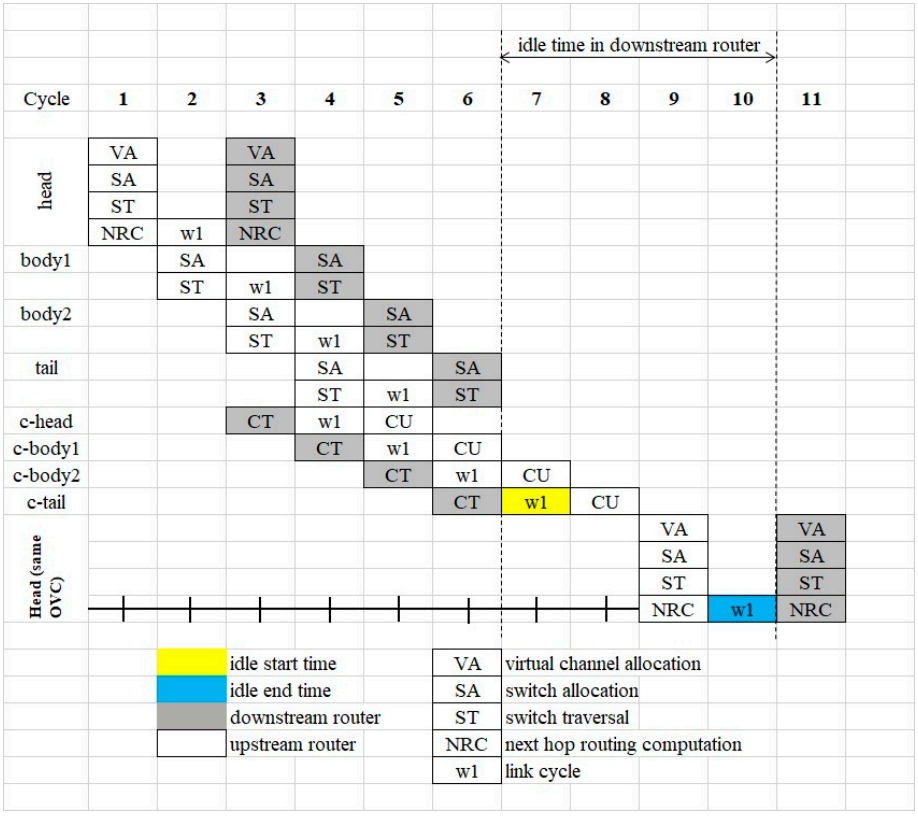

Figure 7. The pipeline of the single stage NoC router, assuming 1 cycle link traversal (W1).

4.2. Hardware Redundancy Technique

The VC status fields are split into two groups according to their role. The first group includes (G, R, O, C) which represent the control of the VC. The second group includes the P field, encompassing the head pointer (PH) and tail pointer (PT) which control the buffer read and write operations, respectively. The hardware redundancy technique is realized by adding two spare registers, Sp1 and Sp2. The former is used as a spare for the first group to replace one of the state fields (G, R, O, or C) when one is faulty. The size of Sp1 depends on the maximum size of state fields (G, R, O, or C). The Sp2 register represents the spare register for the second group, replacing PH or PT when one is faulty. The size of Sp2 depends on the number of buffers in the VC (size of head and tail pointers). The hardware redundancy technique can sustain a single fault in each group. The general scheme of the proposed hardware redundancy circuit, shown in Figure 8, depends on the outputs of the BIST circuit, State1 and State2, which are used to indicate the faulty field that should be replaced with Sp1 and Sp2, respectively. Figure 9 shows the internal design of the BIST and Group2. The Baseline Router VC FSM is the unaltered hardware that changes G, R, O, P, and C according to the VC status and generates the new (next) values of G, R, O, P, and C. The BIST Controller starts the test operation when (reset = 0) or (G = 3 and vc_release = 1) and controls the Pattern Generator, Faults Detection, and States.

4.3. VC-Closing Property

When the number of faults in the VC status fields exceeds the hardware redundancy fault tolerance capability, the VC will malfunction. This occurs when more than one fault affects the same group of VC status fields in the VC. As a result, VC-closing technique is used to close this faulty VC and inform the upstream router to avoid allocating packets to this VC. This is realized by adding a new signal for each VC in each input port (VC_closing_out signal) which goes out from the input port of the downstream router to the output port of the upstream router to prevent the latter from allocating

this faulty VC during the VA stage, as shown in Figure 10. This technique preserves packets transfer through an input port in the event of VC failures until all VCs of that input port fail.

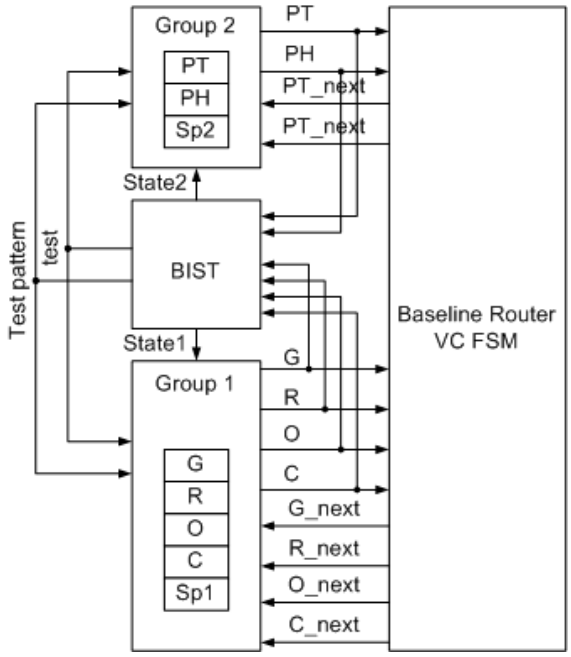

Figure 8. The general scheme of the proposed hardware redundancy circuit.

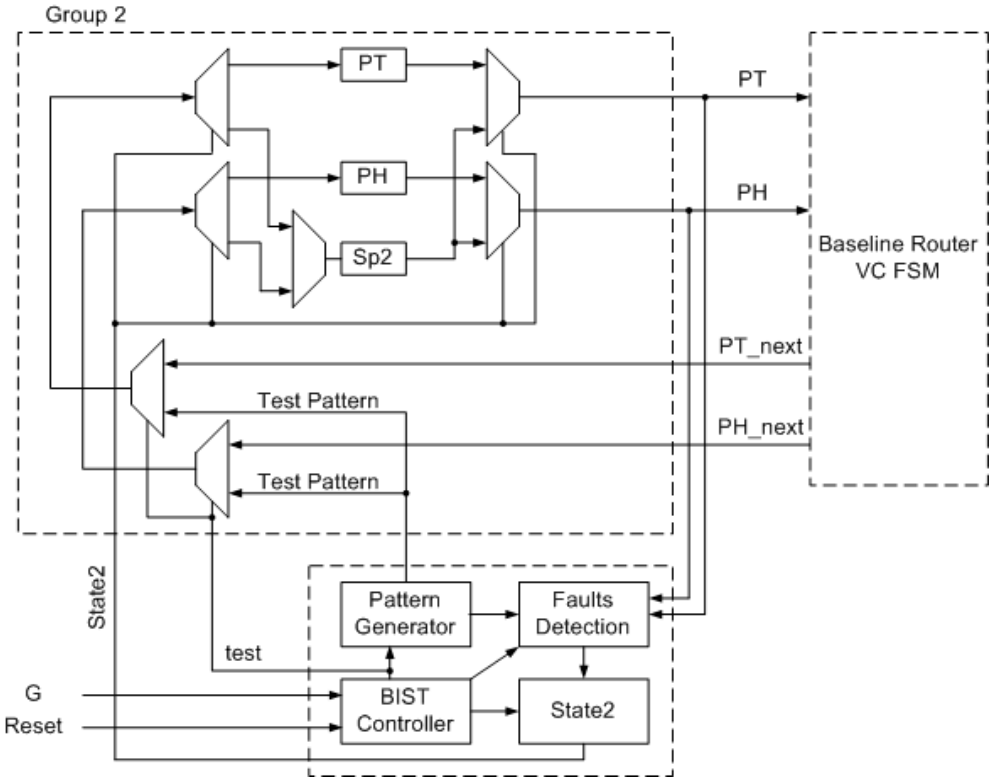

Figure 9. The internal design of the BIST and Group 2 of the general scheme of the proposed hardware redundancy circuit.

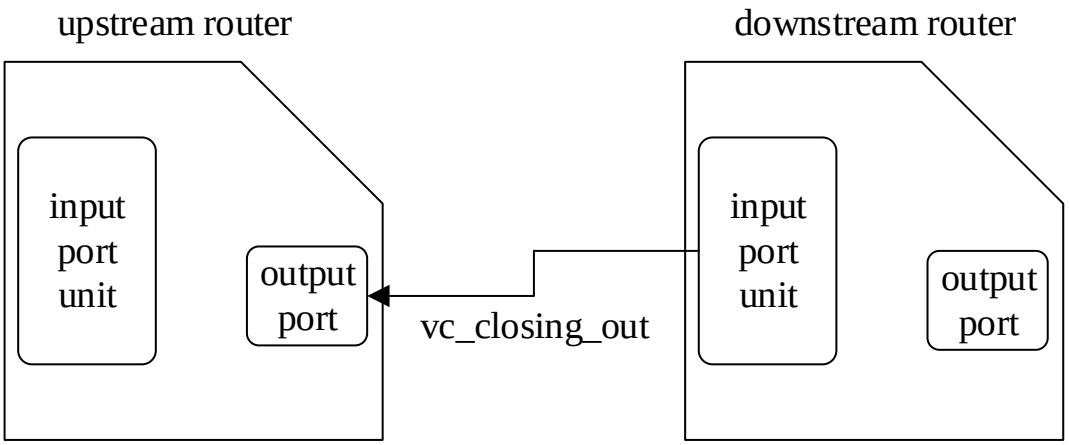

Figure 10. The VC-closing mechanism.

5. Results and Discussion

Our reliable router is compared with the baseline router from delay, area, power, and reliability perspectives. Both routers are developed in Verilog HDL, functionally verified using ModelSim, and then synthesized using Synopsys Design Compiler with 45 nm Nangate's Open Cell library. All analyses were conducted on a router with five input ports and five output ports. Different numbers of virtual channels were considered, namely 2 VCs, 4 VCs, and 8 VCs, with four flits per VC and 32-bit flit size.

5.1. Critical Path Delay Analysis

Figure 11 shows the delay of our reliable router compared to the baseline router for 2 VCs, 4 VCs, and 8 VCs with four flits per VC for each case. The delay is increased by 23%, 9%, and 11% for 2, 4, and 8 VCs, respectively, when our proposed fault detection and tolerance circuitries are added. The critical path delay results were found by repeating the synthesis process while changing the clock period until zero slack is obtained. The virtual channel allocation component in both the baseline and the proposed design represents the critical path, which takes the input virtual channel state fields as input to reserve the output virtual channel and saves the result in the input virtual channel state field O. Due to the additional multiplexers and demultiplexers and their corresponding selection signals the critical path has increased in the proposed design.

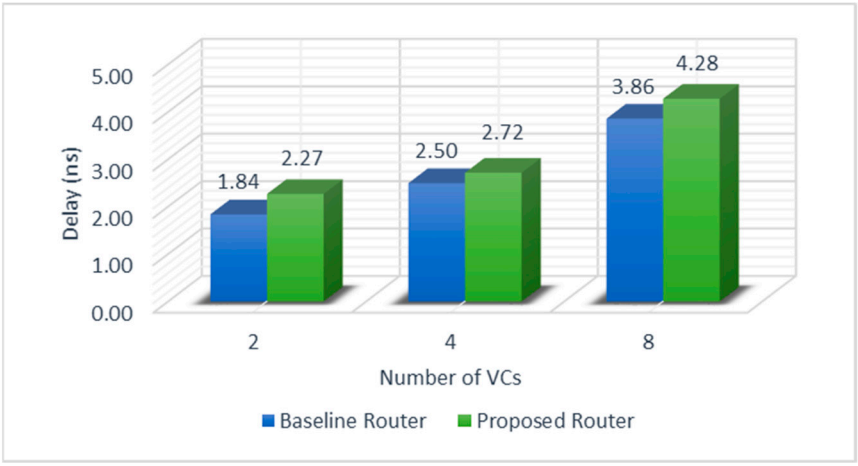

Figure 11. The delay of our reliable router compared to the baseline router for 2VCs, 4VCs, and 8VCs.

5.2. Area Analysis

Figure 12 shows the area overhead of our reliable router compared to the baseline router for 2 VCs, 4 VCs, and 8 VCs at a clock frequency of 426 MHz, 339 MHz, and 222 MHz, respectively. The area of baseline router for 2 VCs, 4 VCs, and 8 VCs is increased by 17%, 17%, and 15%, respectively, when our proposed fault detection and tolerance circuitries are added. Figure 13 shows how the area is partitioned among the different router parts for the baseline and the spare routers for 4 VC case. It is clear how the input port consumes the highest portion of the router area. It can be seen that the input ports are affected by the highest amount of area overhead since they are the target of the proposed design. The output ports also suffer from slight area overhead due to the VC_closing technique. It is also important to indicate that the VC state fields in the baseline router consume an area of 1490 μm^2, which is higher than that of the RC, SA, and ST components.

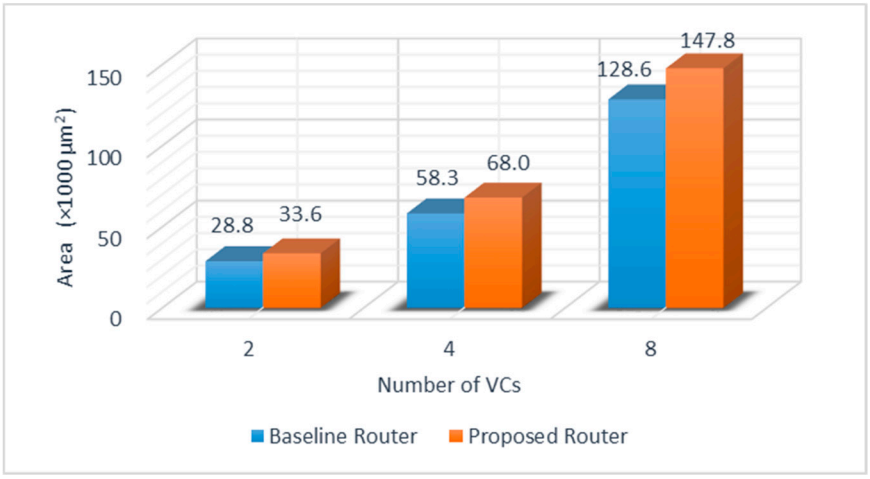

Figure 12. The area overhead of our reliable router compared to the baseline router for 2 VCs, 4 VCs, and 8 VCs.

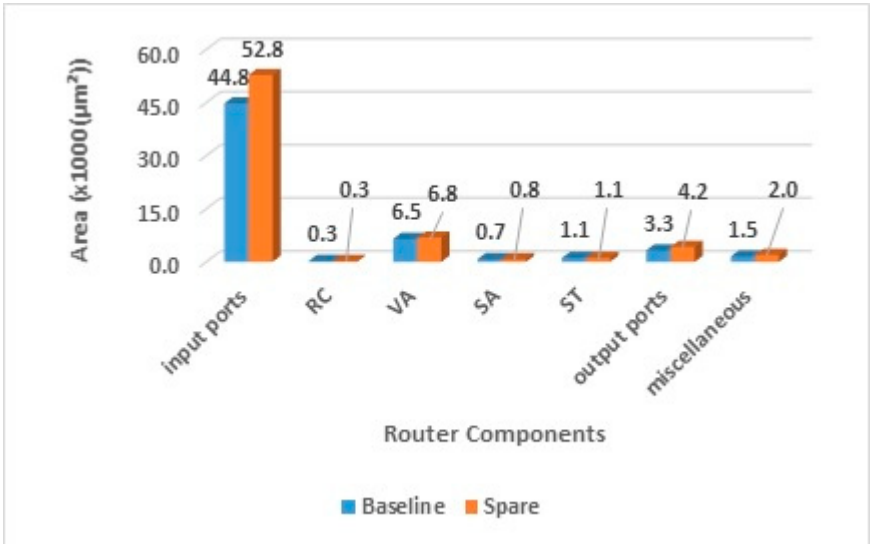

Figure 13. The area of all components of the baseline router and the proposed router.

5.3. Power Analysis

Figure 14 shows the dynamic power consumption of our reliable router compared to the baseline router for 2 VCs, 4 VCs, and 8 VCs at a clock frequency of 426 MHz, 339 MHz, and 222 MHz, respectively. The static power is neglected, as it constitutes less than 1% of the total power consumption.

The dynamic power consumption of the baseline router for 2 VCs, 4 VCs, and 8 VCs are increased by 16%, 11%, and 10%, respectively, due to the fault tolerance circuit and the BIST circuit.

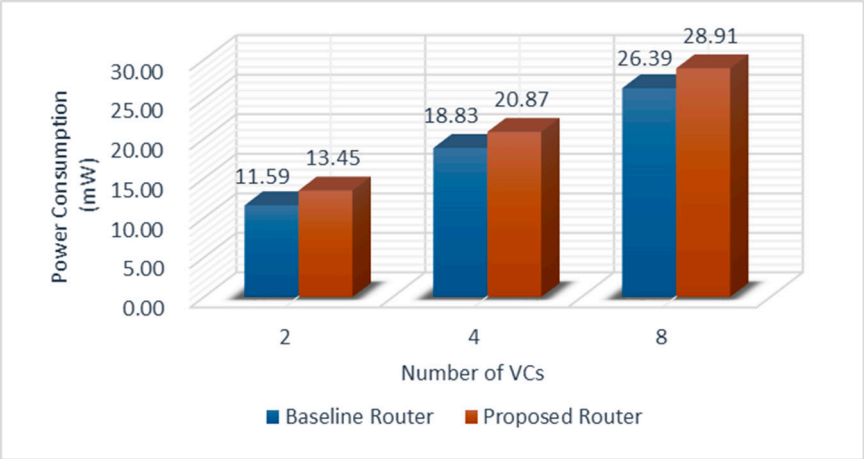

Figure 14. The dynamic power consumption of the proposed reliable router compared to the baseline router for 2 VCs, 4 VCs, and 8 VCs.

5.4. Reliability Improvement

The silicon protection factor (SPF) metric is used to assess the reliability of our proposed reliable NoC router as compared to the baseline router and other state-of-the-art reliable routers. SPF is defined as the ratio of the mean number of faults required to cause a failure and the area overhead acquired due to the fault tolerance circuitry [19]. Our reliable router tolerates only input buffer component faults in NoC router, so other components (RC, VA, SA, and ST) are not tolerated.

$$\text{SPF} = [(\max + \min + 1)/2]/(1+\text{area overhead}) \quad (1)$$

$$\text{area overhead} = [(\text{area of proposed router} - \text{area of baseline})/(\text{area of baseline})] \times 100\% \quad (2)$$

where min is the minimum number of faults to cause a failure, max is the sum of maximum faults tolerated by each component. Since the stages RC, VA, SA, and ST are not tolerated, then the minimum number of faults that cause failure is one, and the maximum number of faults tolerated in each stage is zero, and there is a maximum number of faults tolerated in input buffer stage only. To compute the SPF of the proposed router when the number of VCs is four so the minimum number of faults that made failure in each stage of proposed router (IB, RC, VA, SA, and ST) are (2, 1, 1, 1, and 1), respectively, and the maximum number of faults that are tolerated in each stage of proposed router (IB, RC, VA, SA, and ST) are (55, 0, 0, 0, and 0), respectively. The maximum number of tolerated faults in each port for four VCs is the case when three VCs are closed (having three faults in each), and the fourth VC is active and having two faults. The SPF of the baseline is one in all cases because the minimum number of faults that made failure in the baseline router equals one, the maximum number of faults that can be tolerated in each stage equals zero, and the total area is one. So, from Equations (1) and (2), it can be inferred that the SPF of the baseline router for any number of VCs is one. As a result, the total number of faults for a single port is 11, and for all ports of the 5-ports router, the total maximum number of tolerated faults is 55 faults. Substituting the two values (min = 1 and max = 55) in Equations (1) and (2) infers that the SPF of the proposed router for 4 VCs with area overhead of 0.17 is 24. Similarly, the SPF for 2 VCs and 8 VCs equals 12 and 51, respectively. Figure 15 shows the SPF for our proposed router compared with the baseline router for 2 VCs, 4 VCs, and 8 VCs. The results indicate that for our proposed design the SPF increases when the number of VCs increases. Table 2 shows the comparison of area overhead and SPF for Poluri [24] and Wang [23], and our proposed router. Poluri's [24] SPF resulted from the minimum faults that cause failure in stages RC, VA, SA, or ST namely (2, 4, 2, 2),

and the maximum tolerated faults in all these stages (5 + 15 + 5 + 2). The mean number of faults to cause failure is 15 ([2 + 28]/2) and the normalized area is 1.31, so the SPF equals 11(15/1.31). In the same way, the SPF of Wang [23] resulted from max (IB, RA, VA, SA, and ST) equals 56 (16 + 5 + 20 + 10 + 5), min(4(IB), (1) RC, (2) VA, (2) SA, or (2) ST) equals 1, then the mean number of faults to cause failure is 28.5. Since the normalized area is 1.30, the SPF value becomes 21.9.

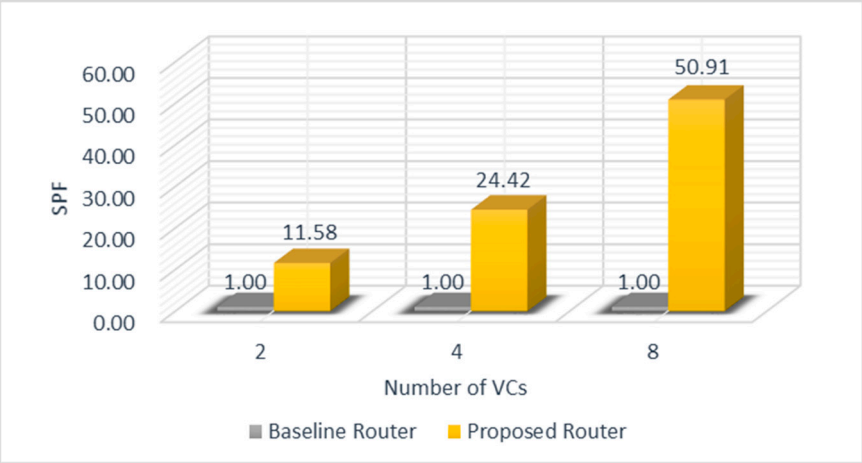

Figure 15. The Silicon Protection Factor for our proposed router compared with the baseline router for 2 VCs, 4 VCs, and 8 VCs.

Table 2. Comparison of area overhead and silicon protection factor (SPF) for Vicis, Shield, Wang, and our proposed router.

Architecture	Area	#Mean Permanent Faults to Cause Failure	Fault Tolerance Methods	SPF
Poluri Router [24]	31%	15	Redundant RC unit, sharing arbiters for VA, bypass path for arbiters of SA, and two paths for a crossbar.	11
Wang Router [23]	30%	28.5	ECC detection and VC closing strategy for input buffer faults, double routing strategy for RC faults, default winner strategy for VA, runtime arbiter selection strategy for SA, and a crossbar with double bypass bus.	21.9
Proposed Router	17%	28.5	BIST detection, VC-closing strategy, and Hardware Redundancy technique for input virtual channel state fields.	24.35

Rather than suggesting protecting only the VC state fields, this work aims to highlight their importance and the possible reliability enhancement when protected. In addition, the proposed input port fault tolerance can be integrated into the other reliable routers to realize a complete, reliable router.

In addition to the SPF factor, we estimate the lifetime enhancement achieved by the proposed input port component compared to the input port component of the baseline 4-stage NoC router using the mean time to Ffailure (MTTF). The main equation of MTTF is:

$$MTTF_{System} = \frac{1}{\frac{1}{\text{MTTF}_1} + \frac{1}{\text{MTTF}_2} + \frac{1}{\text{MTTF}_3}} = \frac{10^9}{FIT1 + FIT2 + FIT3} \quad (3)$$

where 1, 2, 3 are the components of the system; *FIT* is the number of failures in billion hours. The MTTF of the input port component of the baseline router can be given by:

$$MTTF_{baseline\ input\ port} = \frac{10^9}{\text{FIT}_{\text{buffer}} + \text{FIT}_{\text{Group1}} + \text{FIT}_{\text{Group2}}}. \quad (4)$$

where FIT_{buffer} is failure in time of the buffer component; FIT_{Group1} is failure in time of Group1 of VC state fields, and FIT_{Group2} is failure in time of Group2 VC state fields. The MTTF of the input port component of the reliable router is:

$$MTTF_{reliable\ input\ buffer\ component} = \frac{10^9}{\text{FIT}_{\text{buffer}} + \frac{1}{\text{MTTF}_{\text{reliable Group1}}} + \frac{1}{\text{MTTF}_{\text{reliable Group2}}}} \quad (5)$$

Each group in the proposed input port can work normally if the baseline group or the corresponding correction circuitry are fault-free. Accordingly, the MTTF of each group can be given by [23,24]:

$$MTTF_{reliable\ Group1} = \frac{10^9}{\text{FIT}_{\text{Group1}}} + \frac{10^9}{\text{FIT}_{\text{correction Group1}}} + \frac{10^9}{\text{FIT}_{\text{Group1}} + \text{FIT}_{\text{correctionGroup1}}} \quad (6)$$

$$MTTF_{reliable\ Group2} = \frac{10^9}{\text{FIT}_{\text{Group2}}} + \frac{10^9}{\text{FIT}_{\text{correction Group2}}} + \frac{10^9}{\text{FIT}_{\text{Group2}} + \text{FIT}_{\text{correction Group2}}} \quad (7)$$

The FIT values of the fundamental components due to time-dependent dielectric breakdown (TDDB) can be found in [23,24]. Table 3 lists the fundamental components (FC), their corresponding FIT values as given in [23,24], total number of each FC, and the total FIT values in the input port component. Similarly, Tables 4 and 5 present the computed values for Group1 correction circuitry and Group2 correction circuitry, respectively. The FIT of each group correction circuitry is the sum of the FIT of the spare bits and state bits. Substituting the values of Table 3 into Equation (4), the value of MTTF of baseline input port component is found to be (699301) hours. From the values of the three tables (Tables 3–5), we substituted these values in Equations (5)–(7) the MTTF of the reliable input port component is found to be 781,250 h, which is higher than that of the baseline input port.

Table 3. Failures in billion hours (FIT) values of the input port of the baseline 4-stage NoC router.

Fundamental Component (FC)	FIT of FC	#FCs	FIT of FCs
32-bit DFF (BF)	0.5	5 × 4 × 4	1280
11-bit input VC state fields DFF Group1 (G1)	0.5	5 × 4	110
4-bit input VC state fields DFF Group2 (G2)	0.5	5 × 4	40

Table 4. FIT values of Group1 correction circuitry.

Fundamental Component (FC)	FIT of FC	#FCs	FIT of FCs
3-bit DFF (Sp1)	0.5	5 × 4	30
4-bit DFF (State 1)	0.5	5 × 4	40

Table 5. FIT values of Group2 correction circuitry.

Fundamental Component (FC)	FIT of FC	#FCs	FIT of FCs
2-bit DFF (Sp2)	0.5	5 × 4	20
2-bit DFF (State 2)	0.5	5 × 4	20

5.5. Latency Analysis

We considered an 8 × 8 mesh topology and each packet is composed of four flits with 32-bit each. A uniform traffic pattern with Bernoulli injection is used in the simulations. Figure 16 compares three cases: first case is when all VCs are working, the second case is when one VC is closed in the north input port component of the router (y = 4, x = 3), and the last case is when one VC is closed in the west input port component of the router (y = 3, x = 4). It can be seen that the performance degrades when one VC in one router out of 64 routers is closed for both west and north VC closed cases. The west port

has higher performance degradation because in XY routing strategy the west port may send packets to any of the other ports whereas the north port may send to the south and local ports only. This gives an interesting result that requires further analysis, which is out of the scope of this work. Based on the results in the figure it is clear the importance of avoiding the closure of any VC. This was resolved in the proposed design by adding the spare registers (Sp1, Sp2) which allow the router to continue working at full performance in spite of the occurrence of a single fault in each group (Group1, Group2), and the proposed scheme closes a VC only when two faults affect the same group.

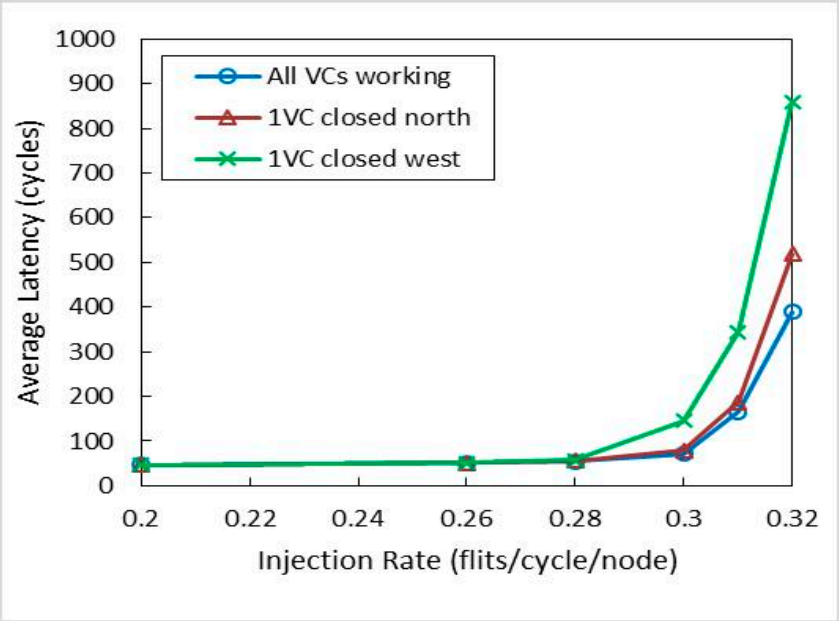

Figure 16. The average latency when closing one VC.

6. Conclusions

A reliable input buffer component in NoC router is proposed in this work. The proposed solution includes fault detection and fault tolerance techniques to tolerate permanent faults in critical fields of the input buffer component called G, R, O, P, and C fields which control the work of virtual channels. Hardware redundancy technique was used to enhance the router reliability with low area, power consumption, and delay overheads with respect to the baseline router. In addition, the proposed router achieved 11% higher reliability than other fault tolerant router designs, represented by the silicon protection factor metric. The protection by hardware redundancy has an important effect on preserving the network performance since one VC closure showed a clear effect on the average packet latency.

Author Contributions: All authors contributed substantially to the paper. Conceptualization, H.M., W.F. and F.R.; Formal analysis, H.M. and W.F.; Funding acquisition, F.R.; Methodology, H.M. and W.F.; Resources, H.M., W.F. and F.R.; Supervision, W.F.; Validation, H.M. and W.F.; Writing—original draft, H.M.; Writing—review and editing, H.M., W.F. and F.R.

Funding: This research was supported in part by GP-IPS grant provided by Universiti Putra Malaysia.

Acknowledgments: The authors wish to acknowledge Intel Malaysia Design Center (MDC) and MIMOS for the donated computer server and for the software tools, respectively.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Singh, S.K.; Mondal, A.J.; Majumder, A. Generation and Performance Evaluation of Reconfigurable Fault Tolerant Routing Algorithm for 2D-Mesh NoC. *Procedia Comput. Sci.* **2015**, *57*, 232–240. [CrossRef]

2. Xu, J.; Wolf, W.; Henkel, J.; Chakradhar, S. A methodology for design, modeling, and analysis of networks-on-chip. In Proceedings of the IEEE International Symposium on Circuits and Systems, Kobe, Japan, 23–26 May 2005; pp. 1778–1781.
3. Kamali, H.M.; Azar, K.Z.; Hessabi, S. DuCNoC: A High-Throughput FPGA-Based NoC Simulator using Dual-Clock Lightweight Router Micro-Architecture. *IEEE Trans. Comput.* **2018**, *67*, 208–221. [CrossRef]
4. Todman, T.J.; Constantinides, G.A.; Wilton, S.J.E.; Mencer, O.; Luk, W.; Cheung, P.Y.K. Reconfigurable computing: Architectures and design methods. *IEE Proc.-Comput. Digit. Tech.* **2005**, *152*, 193–207. [CrossRef]
5. Wang, C.; Hu, W.-H.; Bagherzadeh, N. Scalable load balancing congestion-aware Network-on-Chip router architecture. *J. Comput. Syst. Sci.* **2013**, *79*, 421–439. [CrossRef]
6. Benini, L.; De Micheli, G. Networks on chips: A new SoC paradigm. *Computer* **2002**, *35*, 70–78. [CrossRef]
7. Bjerregaard, T.; Mahadevan, S. A survey of research and practices of Network-on-chip. *ACM Comput. Surv.* **2006**, *38*, 1. [CrossRef]
8. Vangal, S.; Howard, J.; Ruhl, G.; Dighe, S.; Wilson, H.; Tschanz, J.; Finan, D.; Iyer, P.; Singh, A.; Jacob, T.; et al. An 80-Tile 1.28TFLOPS Network-on-Chip in 65nm CMOS. In Proceedings of the 2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers, San Francisco, CA, USA, 11–15 February 2007; pp. 98–99.
9. Dally, W.; Towles, B. *Principles and Practices of Interconnection Networks*; Morgan Kaufmann: Burlington, MA, USA, 2004.
10. Dang, K.N.; Ben Ahmed, A.; Tran, X.-T.; Okuyama, Y.; Abdallah, A. Ben A comprehensive reliability assessment of fault-resilient network-on-chip using analytical model. *IEEE Trans. Very Large Scale Integr. Syst.* **2017**, *25*, 3099–3112. [CrossRef]
11. Pande, P.P.; Grecu, C.; Ivanov, A.; Saleh, R.; De Micheli, G. Design, synthesis, and test of networks on chips. *Des. Test Comput. IEEE* **2005**, *22*, 404–413. [CrossRef]
12. Sui, P.-H.; Wang, S.-D. An improved algorithm for fault-tolerant wormhole routing in meshes. *IEEE Trans. Comput.* **1997**, *46*, 1040–1042.
13. Ziegler, J.F. Terrestrial cosmic rays. *IBM J. Res. Dev.* **1996**, *40*, 19–39. [CrossRef]
14. Vrudhula, S.B.K.; Blaauw, D.; Sirichotiyakul, S. Estimation of the likelihood of capacitive coupling noise. In Proceedings of the 39th annual Design Automation Conference, New Orleans, LA, USA, 10–14 June 2002; pp. 653–658.
15. Kuhn, K.J. Reducing variation in advanced logic technologies: Approaches to process and design for manufacturability of nanoscale CMOS. In Proceedings of the 2007 IEEE International Electron Devices Meeting, Washington, DC, USA, 10–12 December 2007; pp. 471–474.
16. Hu, C.-K.; Rosenberg, R.; Rathore, H.S.; Nguyen, D.B.; Agarwala, B. Scaling effect on electromigration in on-chip Cu wiring. In Proceedings of the IEEE 1999 International Interconnect Technology Conference, San Francisco, CA, USA, 26 May 1999; pp. 267–269.
17. Wu, E.; Sune, J.; Lai, W.; Nowak, E.; McKenna, J.; Vayshenker, A.; Harmon, D. Interplay of voltage and temperature acceleration of oxide breakdown for ultra-thin gate oxides. *Solid. State. Electron.* **2002**, *46*, 1787–1798. [CrossRef]
18. Radetzki, M.; Feng, C.; Zhao, X.; Jantsch, A. Methods for fault tolerance in networks-on-chip. *ACM Comput. Surv.* **2013**, *46*, 1–38. [CrossRef]
19. Constantinides, K.; Plaza, S.; Blome, J.; Zhang, B.; Bertacco, V.; Mahlke, S.; Austin, T.; Orshansky, M. BulletProof: A defect-tolerant CMP switch architecture. In Proceedings of the Twelfth International Symposium on High-Performance Computer Architecture, Austin, TX, USA, 11–15 February 2006; pp. 5–16.
20. Fick, D.; DeOrio, A.; Hu, J.; Bertacco, V.; Blaauw, D.; Sylvester, D. Vicis: A reliable network for unreliable silicon. In Proceedings of the 46th Annual Design Automation Conference, San Francisco, CA, USA, 26–31 July 2009; pp. 812–817.
21. Das, C.R.; Yousif, M.S.; Narayanan, V.; Park, D.; Nicopoulos, C.; Kim, J.; Das, C.R.; Yousif, M.S.; Narayanan, V.; Park, D.; et al. A Gracefully Degrading and Energy-Efficient Modular Router Architecture for On-Chip Networks. In Proceedings of the 33rd International Symposium on Computer Architecture (ISCA'06), Boston, MA, USA, 17–21 June 2006; pp. 4–15.
22. Poluri, P.; Louri, A. An improved router design for reliable on-chip networks. In Proceedings of the 2014 IEEE 28th International Parallel and Distributed Processing Symposium, Phoenix, AZ, USA, 19–23 May 2014; pp. 283–292.

23. Wang, L.; Ma, S.; Li, C.; Chen, W.; Wang, Z. A high performance reliable NoC router. *Integr. VLSI J.* **2017**, *58*, 583–592. [CrossRef]
24. Poluri, P.; Louri, A. Shield: A reliable network-on-chip router architecture for chip multiprocessors. *IEEE Trans. Parallel Distrib. Syst.* **2016**, *27*, 3058–3070. [CrossRef]