RZ 3443 (# 93738) 08/12/02
Computer Science 58 pages

Research Report

Worst-Case Resequence Queue Size Evaluation for a PPS/Buffered-Crossbar Architecture

Ilias Iliads

IBM Research
Zurich Research Laboratory
8803 Rüschlikon
Switzerland

Worst-Case Resequence Queue Size Evaluation for a PPS/Buffered-Crossbar Architecture

I. Iliadis

IBM Research, Zurich Research Laboratory, 8803 Rüschlikon, Switzerland

Abstract

An analytical method is presented for the evaluation of the worst-case resequence queue size and the output buffer size required by a combined PPS/buffered-crossbar architecture. The impact of the load balancing mechanism is assessed by considering a static mechanism based on a round-robin scheme and a dynamic mechanism based on a state-dependent scheme. The influence of the dedicated and shared output buffer arrangements is also investigated.

Keywords

Parallel packet switch, buffered-crossbar switch fabric, resequence queue, output buffer sizing.

I. Introduction

The Parallel Packet Switch (PPS) architecture was proposed in [1] in order to cope with the problem arising when line rates run faster than the switch fabric. This paper deals with the issue of evaluating the resequence queue size and the corresponding output buffer size required in the context of a PPS architecture based on an NxN buffered crossbar switch, such as the distributed packet routing switch [2]. This architecture provides the capability of switching variable length frames. It is also assumed that the system supports a number of P priorities. Each of the N input and N output adaptors has a fixed number of n subports. For illustration purposes we consider $N = 64$ and $m = 4$ as shown in Figure 1. Each subport is running at a nominal speed of one (e.g. at OC-192 or at OC-768). The round-trip time (RTT) between the switch and the adaptors is assumed to be negligible. However, the results obtained are also likely valid in the case of non-negligible RTTs. This is because the impact of the RTT on the events of the various processes considered in this work is confined only to a shift of the time instances at which these events take place. This in turn does not affect the magnitude of these processes and, consequently, does not affect the

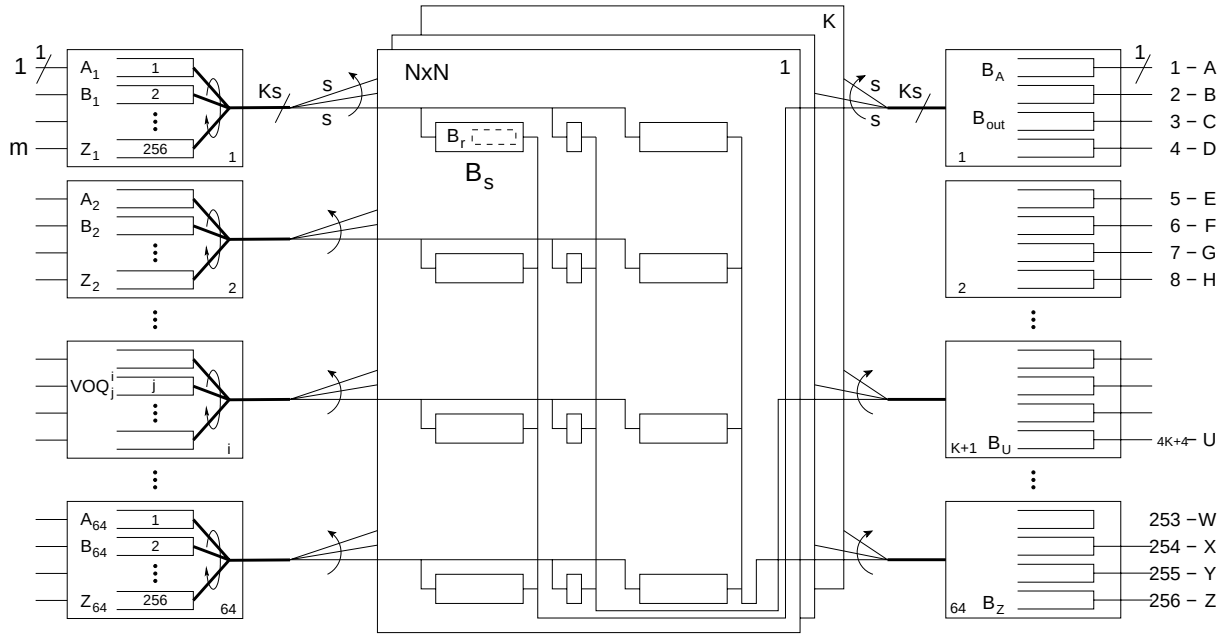

Fig. 1. The PPS/buffered-crossbar switch.

obtained results. Variable length frames are segmented at the ingress into fixed length segments which are subsequently transmitted through the switch via fixed-size packets. In the following a space unit is taken to be equal to the size of such a packet. Although by definition all packets of a frame arrive at the same subport and depart from the same output subport, the reverse association cannot be established at the switch level. This is because the only information pertaining to packets is their incoming and outgoing subports, as well as their priority. Consequently, a *flow* of successive packets arriving at a given input subport and destined to a given output subport may belong to different frames which cannot be recognized by the switch. The maximum number of packets corresponding to a flow is denoted by $F_{\max}$. If the switch can recognize frames, then the analysis and results presented hold by replacing the term flow with that of frame. Therefore, in that case, $F_{\max}$ denotes the maximum number of packets corresponding to a frame. Time is divided into units called time slots with duration equal to the fixed transmission time of a packet at the subport link speed.

Packets are transferred from the input to the output adaptors over multiple switching planes as shown in Figure 1. We consider a number of K ($K \geq 2$) planes each of which is an NxN distributed packet routing switch with a buffered crossbar architecture and a port speed of s packets per time slot, as shown in Figure 1. The *speedup* or *expansion factor* of the configuration consid-

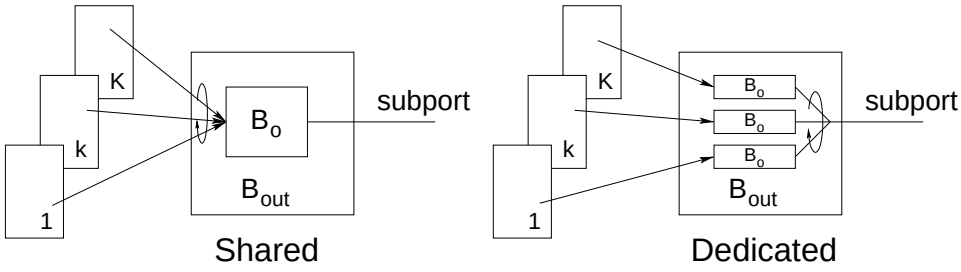

Fig. 2. Output buffer arrangement.

ered is given by

$$n = \frac{m}{Ks}\,. \tag{1}$$

At each input–output pair corresponds a buffer, of size of B_s units, which is shared among the different priorities. Let B_r be the maximum buffer occupancy with packets of the highest priority. Each input adaptor contains Nm (256) Virtual Output Queues (VOQs) corresponding to the Nm (256) output subports. Each VOQ can be fed at a maximum rate of n (4) packets per slot. VOQs are served according to priorities, and according to a round-robin scheme within a priority. In one time slot, the *plane load balancing scheme* is assumed to have the capability to dispatch s packets per plane[1], i.e. up to Ks packets to the corresponding planes as shown in Figure 1.

Two plane load balancing schemes are considered: a static one based on a round-robin scheduler and a dynamic one based on state-dependent information. These two schemes are described in detail and analyzed in Sections II-A.1 and II-A.2, respectively. The *plane server scheme* is assumed to operate on a round-robin fashion. Each output adaptor contains n (4) buffers, one for each subport. Each output buffer has a size of $B_{\rm out}$ units and is assumed to be either shared among the planes, or split to K dedicated buffers, one for each plane. Let also $B_{\rm o}$ denote the size of the shared buffer in the former case and the size of the dedicated buffer in the latter case, as shown in Figure 2. Then, for the two cases, it holds that $B_{\rm out} = B_{\rm o}$ and $B_{\rm out} = KB_{\rm o}$, respectively. In the case of shared output buffer, , the *plane server scheme* can transfer in one time slot s packets from each plane to any given output adaptor, implying a maximum rate of Ks packets per time slot.

According to the scheme described above, packets are transferred from the input to the output adaptors over multiple switching planes. As a result, packets of a flow may arrive at an output adaptor in a different order from the one in which they originally entered the input adaptor. As FIFO delivery is required, out-of-sequence packets must wait at the output adaptor to be put back

[1] In the case where $s < 1$, there can be at most one packet dispatched every $1/s$ slots.

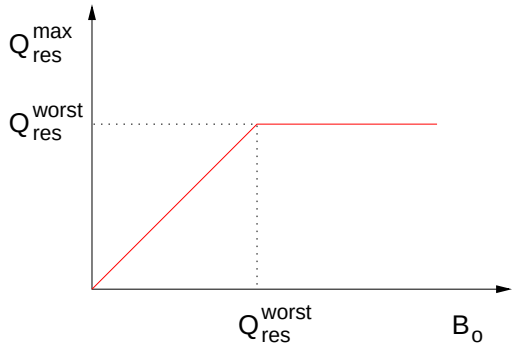

Fig. 3. Maximum resequence queue size vs. output buffer size.

into proper sequence. Therefore, in addition to queueing and transmission delay, out-of-sequence packets incur resequencing delay due to the reordering of the output stream at the receiving output adaptor. Out-of-sequence packets stored in an output buffer constitute a virtual *resequence queue*. It can be shown that the expected number of out-of-sequence packets in an output buffer (or equivalently, the expected resequence queue size) is proportional to the mean resequencing delay [3]. Let also $Q_{\rm res}^{\rm max}$ denote the maximum resequence queue size in output buffer $B_{\rm o}$. Clearly, it holds that $Q_{\rm res}^{\rm max} \leq B_{\rm o}$. For small values of $B_{\rm o}$ we expect $Q_{\rm res}^{\rm max}$ to be equal to $B_{\rm o}$ as the output buffer can be filled with out-of-sequence packets. In the case of dedicated buffers, it turns out that all the K output buffers can simultaneously be filled with out-of-sequence packets. The fact that no further departures are possible, implies that no subsequent arrivals are possible, and therefore the system is deadlocked. However, for sufficiently large values of $B_{\rm o}$ we expect $Q_{\rm res}^{\rm max}$ to be less than $B_{\rm o}$ and in fact converge to the worst-case resequence queue size, denoted by $Q_{\rm res}^{\rm worst}$ as shown in Figure 3. Clearly, the minimum output buffer size required for a safe operation is equal to the worst-case resequence queue size incremented by one packet, i.e. $B_{\rm o} \cong Q_{\rm res}^{\rm worst}$. The objective of the work presented here is the evaluation of this value as well as of the range of the switch port speed s in which this value is valid. The analysis is conducted by first considering a sufficiently large switch speed (e.g. equal to the subport speed by setting $s = 1$), and then determining the minimum value of the switch speed, $s_{\rm min}$, for which the analysis is still valid. The value $s_{\rm min}$ is derived from the maximum packet rate observed at any of the incoming and outgoing switch links.

II. Worst Case of Resequence Queue Size

We describe here the basic principle governing the assessment of worst case of resequence queue size at an output adaptor. Clearly, for a typical sequence of packets $i, i+1, i+2, \ldots$, belonging

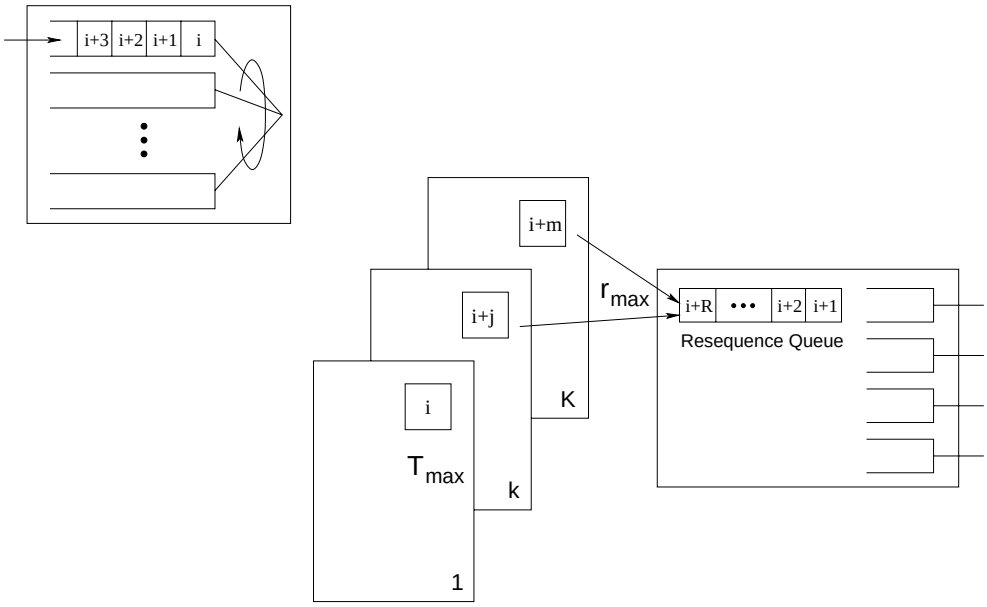

Fig. 4. Worst-case resequence queue size.

to a flow as depicted in Figure 4, the resequence queue size at the corresponding output buffer is maximized when the following two events occur:

(a) Packet i of the flow spends the maximum possible amount of time $T_{\max}$ in its plane, say plane 1.

(b) During this period, a maximum number of subsequent packets of this flow arrive at the corresponding output adaptor through the remaining planes. Let $r_{\max}$ denote the corresponding maximum possible packet transfer rate.

Then the worst-case resequence queue size is given by,

$$Q_{\rm res}^{\rm worst} = r_{\max}\, T_{\max}\,. \tag{2}$$

In Figure 4 it is shown that packets $i+1,\ldots,i+R$ have already arrived and stored at the output (shared) buffer of the corresponding destination adaptor forming the resequence queue while waiting for packet i to arrive. Note that in the case of dedicated buffers, the resequence queue is fed by a single plane. Also the sequence $i+1,\ldots,i+R$ may not necessarily contain all consecutive packets, as some packets $i+j$, with $1<j<R$ may have been dispatched to plane 1 such that are residing behind packet i.

It is assumed without loss of generality that the packets of the flow causing the worst case of resequence arrive at the first input subport and they are destined to the first output subport

(subport 1-A). Therefore, the packets of this flow (indicated by the red color in the figures to follow) are transmitted through the VOQ A1. Depending on the priority of the packets, two cases are considered: high priority and low priority.

A. High Priority

All the packets of the flow are assumed to have the highest priority. Clearly, the worst case of resequence is obtained by considering only packets of flows having the highest priority. Furthermore, as it will be shown below, during the period that packet i of the flow spends in plane 1, the output buffer corresponding to subport A and fed by plane 1 is full. Also planes 2 through K have enough red flow traffic to sustain a minimum rate of one packet every K slots. Let us now consider in more detail the issue of the output buffer arrangement. In the case of a shared output buffer with a round-robin plane server scheme, at each time slot, one packet is transferred to the output buffer, implying that each plane dispatches one packet to the output buffer every K time slots (see Figure 5, first column). On the other hand, in the case of dedicated buffers, at each time slot one packet is transmitted to the output subport taken from the output buffers in a round-robin fashion. As long as the output buffers are not empty, this implies that plane 1 dispatches one packet to the output buffer every K time slots (see Figure 5, second and third column). From the above it is evident that both, the dedicated output buffer system and the shared output buffer system result in

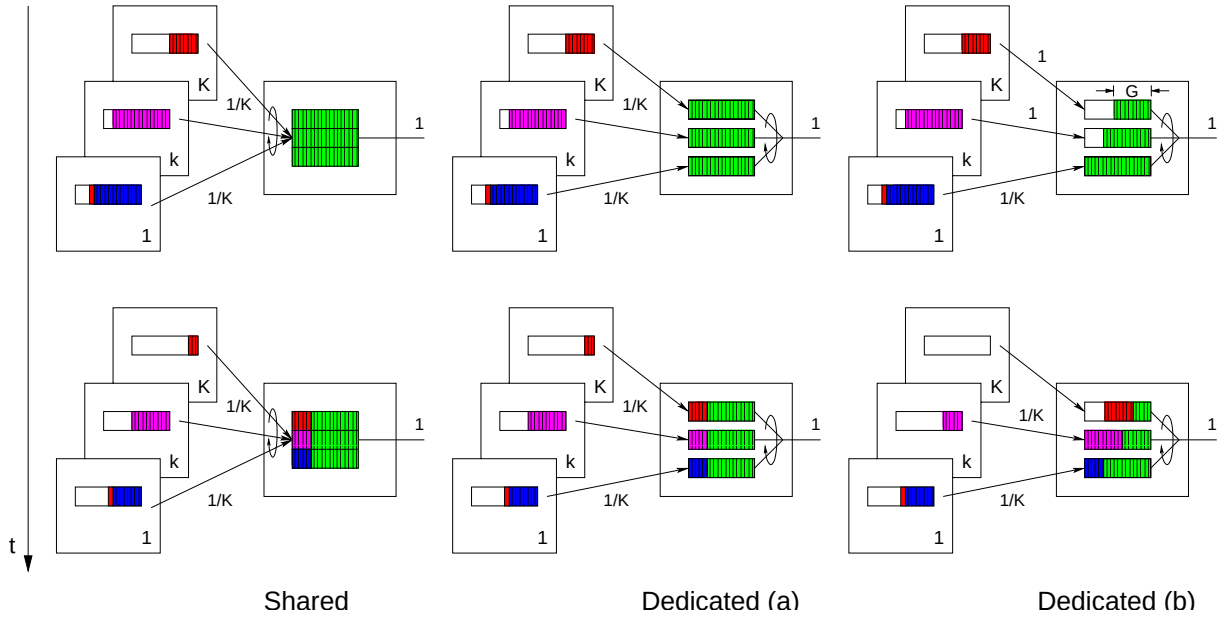

Fig. 5. Shared vs. dedicated output buffer arrangement.

the same amount of time $T_{\max}$ that the red packet i will spend in plane 1. In particular, in the case of dedicated output buffers, this holds if, and only if, the output buffers are never empty during this period. Consequently, focusing on a particular output buffer, and assuming an initial buffer occupancy of G packets as shown in Figure 5, third column, the following condition should be satisfied:

$$G \geq \frac{T_{\max}}{K} \,. \tag{3}$$

Remark 1. If all dedicated output buffers are full, the dedicated output buffer system and the shared output buffer system behave identically, as demonstrated in Figure 5 (first and second column).

Theorem 1: For a switch port speed greater than the inverse of the number of planes, i.e. $s > 1/K$, the total worst-case resequence queue size in the case of dedicated output buffers per plane is larger or equal to the equivalent queue size in the case of a shared output buffer, i.e.

$$Q_{\rm res}^{\rm worst}(\text{total dedicated}) \geq Q_{\rm res}^{\rm worst}(\text{shared}) \qquad \text{for} \quad Ks > 1 \,. \tag{4}$$

Proof: Remark 1 implies that the total worst-case resequence queue size in the case of dedicated output buffers per plane is at least as much as the equivalent queue size in the case of a shared output buffer. On the other hand, in the case of a shared output buffer, the transfer rate of packets out of each plane is one packet every K time slots. However, in the case, of dedicated output buffers, the transfer rate of packets out of each plane (excluding plane 1) can be higher, namely one packet every $1/s$ time slots, for $s > 1/K$. This is depicted in the third column of Figure 5 where the output buffers corresponding to planes 2 through K are not full (and at the same time not empty). Consequently, this implies a potentially larger resequence queue size. ■

Note that as the resequenced packets of a given flow are stored only in the $K-1$ dedicated output buffers, it holds that

$$Q_{\rm res}^{\rm worst}(\text{total dedicated}) = (K-1)\, Q_{\rm res}^{\rm worst}(\text{dedicated}) \,, \tag{5}$$

where that last term of the right hand side represents the worst-case resequence queue size per dedicated output buffer.

Corollary 1: For a switch port speed greater than the inverse of the number of planes, i.e. $s > 1/K$, the minimum output buffer sizes required in the cases of dedicated output buffers and a

shared output buffer satisfy the following inequality,

$$B_{\rm out}(\text{dedicated}) \geq \frac{K}{K-1} B_{\rm out}(\text{shared}) \qquad \text{for} \quad Ks > 1\,. \tag{6}$$

Proof: From (4) and (5) it follows that

$$Q_{\rm res}^{\rm worst}(\text{dedicated}) \geq \frac{Q_{\rm res}^{\rm worst}(\text{shared})}{K-1} \qquad \text{for} \quad Ks > 1\,. \tag{7}$$

In the case of dedicated output buffers, it holds that $B_{\rm out} = KB_{\rm o}$ and $B_{\rm o} \cong Q_{\rm res}^{\rm worst}(\text{dedicated})$. Thus, (7) yields

$$B_{\rm out}(\text{dedicated}) > \frac{K}{K-1} Q_{\rm res}^{\rm worst}(\text{shared}) \tag{8}$$

Eq. (6) follows from (8) by recalling that $B_{\rm out}(\text{shared}) \cong Q_{\rm res}^{\rm worst}(\text{shared})$. ■

A.1 A Round-Robin Load Balancing Scheme

Here we consider the case of a static round-robin load balancing scheme. According to this scheme, packets are transferred from a given input adaptor to the various planes in a cyclic manner, regardless of the occupancy of the buffers in the various planes.

Event (a) translates to the following hot spot scenario: at the instant when packet i is dispatched to plane 1, buffers $B_{i,1}$ ($i = 1, \ldots, 64$) of the first plane corresponding to the first output adaptor are full. This is indicated in Figure 6 with the blue packets all destined to the first output adaptor, and in particular to subport A. Furthermore, the output buffer corresponding to subport A and fed by plane 1 is full. As discussed in Section II-A, the rate that it is fed from plane 1 is equal to one packet every K slots, because of traffic present at the remaining planes which is destined to subport A. This scenario applies in both cases of a dedicated output buffer system and a shared output buffer system, and ensures that packet i will spend the maximum possible amount of time, $T_{\rm max}$, in plane 1, given by

$$T_{\rm max} = KNB_r\,. \tag{9}$$

In the case of dedicated output buffers, the above equation holds if, and only if, the output buffers are never empty during this period. Substituting (9) into (3) yields the following condition

$$G \geq NB_r\,. \tag{10}$$

Event (b) in turn requires that, during this period, a maximum number of packets of the flow under consideration arrive at the corresponding output adaptor through the remaining planes. Furthermore, they should be subsequent to packet i, so that they are stored in the resequence queue and the packet transfer rate is the maximum possible. This is achieved if the remaining planes are practically empty and they are only fed by packets of the flow under consideration, as depicted in Figure 6. Here, however, there is a difference between the two output buffer arrangements expressed by the Theorem 1.

Before proceeding, the following issue should be addressed. Is it possible, under a normal switch operation, to arrive in the scenario described above, whereby the loading asymmetry between plane 1 and any of the remaining planes is so extreme? Note that the plane load balancing scheme ensures that the load is equally distributed among the planes. Consequently, at first glance, this scenario

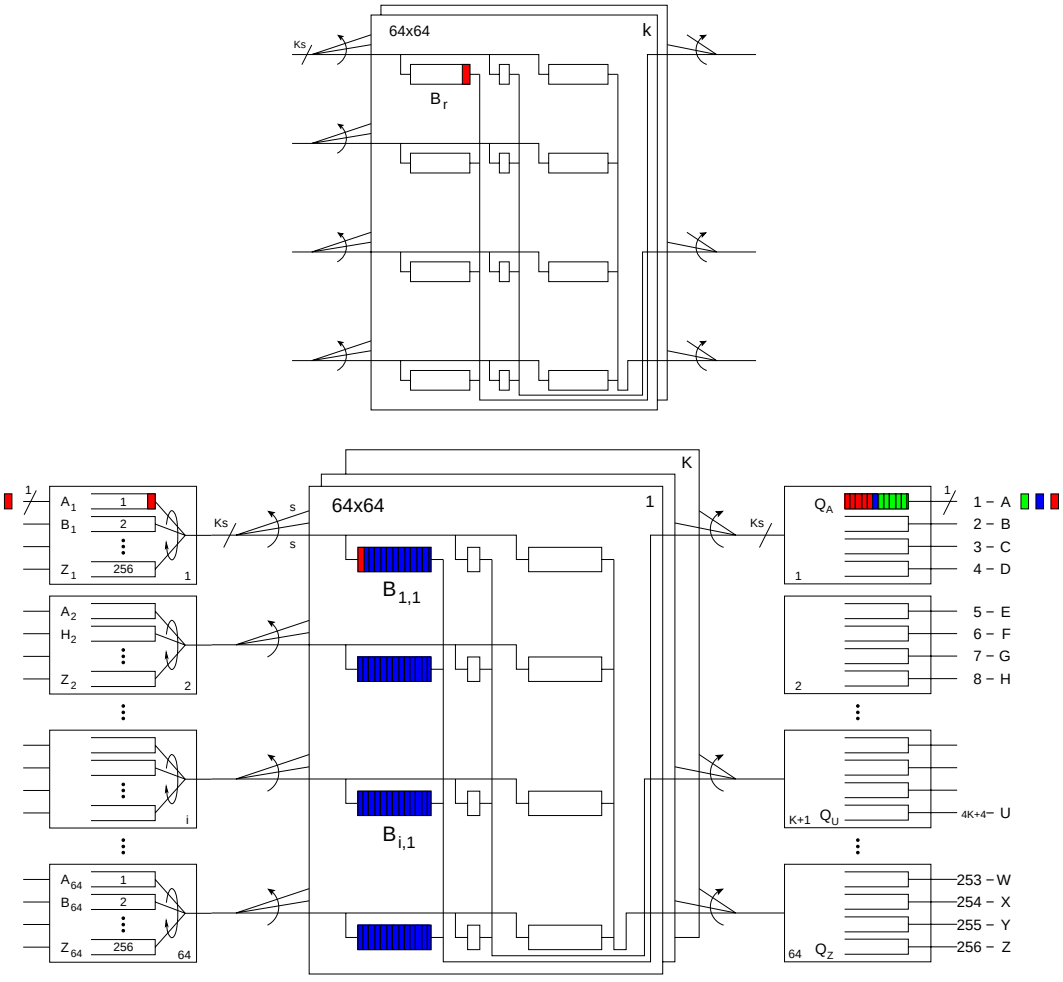

Fig. 6. Worst-case resequencing for a PPS/buffered-crossbar switch configuration.

where all the blue packets are residing in one of the planes seems rather unlikely. Nevertheless, in Appendix A it is shown that this scenario is feasible when the load balancing scheme is round-robin. In Section II-A.2, however, it is shown that this scenario is no longer possible in the case of the state-dependent load balancing scheme.

The maximum resequence queue size and the corresponding minimum output buffer size required are now given by the following theorems.

Theorem 2: In the case of a shared output buffer, for a switch port speed s in the range $s > s_{\min} = \frac{1}{K}$, and for any number of subports n, the maximum resequence queue size is given by

$$Q_{\text{res}}^{\text{worst}}(\text{shared}) = (K-1)NB_r \qquad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,. \tag{11}$$

Proof: To each blue packet arriving at the shared output buffer correspond $K-1$ red packets arriving at this buffer through the remaining of the planes. As all these packets have a higher sequence number than packet i, they need to be stored in the resequence queue. Given that a number of NB_r blue packets will be transferred before packet i to the shared output buffer, the maximum resequence queue size is given by (11). According to Remark 5 of Appendix A, the above equation holds for any value of the switch port speed s in the range $s > s_{\min} = \frac{1}{K}$. Also according to Remark 6 of Appendix A, the above equation holds for any number of subports. ■

Theorem 3: In the case of dedicated output buffers, the maximum resequence queue size for a dedicated output buffer is given by

$$Q_{\text{res}}^{\text{worst}}(\text{dedicated}) = \begin{cases} (N + \dfrac{N-1}{K-1})B_r & \text{for} \quad m = 1 \quad \text{and} \quad s \geq s_{\min} = \dfrac{1}{K}\left[1 + \dfrac{N-1}{N(K-1)}\right] \\ \dfrac{K}{K-1}\,NB_r & \text{for} \quad m \geq 2 \quad \text{and} \quad s \geq s_{\min} = \dfrac{1}{K-1}\,. \end{cases} \tag{12}$$

Proof: See Appendix B. ■

Substituting (12) into (5) yields

$$Q_{\text{res}}^{\text{worst}}(\text{total dedicated}) = \begin{cases} (KN-1)B_r & \text{for} \quad m = 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\left[1 + \frac{N-1}{N(K-1)}\right] \\ KNB_r & \text{for} \quad m \geq 2 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K-1}\,. \end{cases} \tag{13}$$

Remark 2. From Eqs. (11) and (13) it follows that $Q_{\text{res}}^{\text{worst}}(\text{total dedicated}) > Q_{\text{res}}^{\text{worst}}(\text{shared})$, $\forall\, K \geq 2\,, \forall\, N \geq 2\,$, which is consistent with (4).

Theorem 4: The minimum output buffer size required is given by the following expressions.

$$B_{\text{out}}(\text{shared}) > (K-1)NB_r \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,. \tag{14}$$

$$B_{\text{out}}(\text{dedicated}) > \begin{cases} \dfrac{K}{K-1}\,(KN-1)B_r & \text{for} \quad m = 1 \quad \text{and} \quad s \geq s_{\min} = \dfrac{1}{K}\left[1 + \dfrac{N-1}{N(K-1)}\right] \\ \dfrac{K^2}{K-1}\,NB_r & \text{for} \quad m \geq 2 \quad \text{and} \quad s \geq s_{\min} = \dfrac{1}{K-1}\,. \end{cases} \tag{15}$$

Proof: See Appendices A and B. ■

A.2 A State-Dependent Load Balancing Scheme

Here we consider a dynamic state-dependent load balancing scheme. According to this scheme, packets are transferred from a given input adaptor to the planes for which the corresponding output buffers are the least loaded. This is the equivalent of the Join the Shortest Queue (JSQ) policy. More specifically, a packet that arrived at input adaptor i and destined to output adaptor j is dispatched to the plane for which the occupancy of the corresponding buffer $B_{i,j}$ is the lowest among the buffers $B_{i,j}$ of all the K planes. In the case of equal least occupancies, the packets are transferred to the planes in a cyclic manner, i.e. in a round-robin fashion.

Clearly this state-dependent load balancing scheme limits significantly the degree of load asymmetry between planes. Consequently, event (a) can no longer be mapped to the hot spot scenario described in Section II-A.1. It now translates to the following hot spot scenario: at the instant when packet i is dispatched to plane 1, only buffer $B_{1,1}$ is full, whereas buffers $B_{i,1}$ ($i = 2, \ldots, 64$) of the first plane corresponding to the first output adaptor are no longer full as indicated in Figure 6, but now each contains one packet. Furthermore, each of the remaining planes can sustain a minimum rate of one packet every K slots destined to subport A. This scenario ensures that packet i will spend the maximum possible amount of time in plane 1. It turns out that the amount of time that the resequencing takes place is given by

$$T_{\max} = KN\,. \tag{16}$$

As discussed in Section II-A, the above relation holds in both cases of a dedicated and a shared output buffer system. In the case of dedicated output buffers, the above equation holds if, and only if, the output buffers are never empty during this period. Substituting (16) into (3) yields the following condition

$$G \geq N\,. \tag{17}$$

Similarly to Section II-A.1, event (b) in turn implies that each of the remaining planes can sustain a minimum rate of one packet every K slots of the flow under consideration destined to subport A.

The maximum resequence queue size and the corresponding minimum output buffer size required are now given by the following theorems.

Theorem 5: In the case of a shared output buffer, for a switch port speed s in the range $s > s_{\min} = \frac{1}{K}$, and for any number of subports m, the maximum resequence queue size is given by

$$Q_{\text{res}}^{\text{worst}}(\text{shared}) = (K-1)(N-1) \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,. \tag{18}$$

Proof: The proof is similar to that of Theorem 2 and it is therefore omitted. ■

Theorem 6: The minimum output buffer size required is given by the following inequalities.

$$B_{\text{out}}(\text{shared}) > (K-1)(N-1) \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,, \tag{19}$$

and

$$B_{\text{out}}(\text{dedicated}) > K(N-1) \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,. \tag{20}$$

Proof: Eqs. (19) and (20) follow directly from (18) and (6). ■

B. Low Priority

The worst case for the resequence queue size arises in the case where the frame arriving has a low priority, i.e. a priority which is not the highest. In this case event (a) translates to the following scenario: packet i is dispatched to a plane and stays there indefinitely because it is always preempted by traffic of a higher priority. Event (b) in turn implies that the subsequent packets of the frame are transferred to the remaining planes and from there to the first output adaptor. In Appendix D it is shown that this scenario is feasible. It applies to both cases of static and dynamic plane load

balancing. The maximum resequence queue size and the corresponding minimum output buffer size required are now given by the following theorems.

Theorem 7: For a switch port speed s in the range $s > s_{\min} = \frac{1}{K}$, and for any number of subports m, the maximum resequence queue size is given by

$$Q_{\rm res}^{\rm worst}(\text{shared}) = Q_{\rm res}^{\rm worst}(\text{dedicated}) = (N-K)B_r(F_{\max}-1) \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,. \tag{21}$$

Proof: In Appendix D it is shown that the number of packets that need to be resequenced at the output adaptor grows constantly up to the maximum frame size reduced by one for the initially blocked packet, i.e. $F_{\max} - 1$ space units. The number of blocked packets at each crosspoint can be as high as B_r, and the number of such crosspoints for a given output subport can be as high as $N - K$. Consequently, the maximum resequence queue size is given by (21). According to Remark 13 of Appendix D, the above equation holds for any value of the switch port speed s in the range $s > s_{\min} = 1/K$. Also according to Remarks 10 and 11 of Appendix D it holds in both cases of a shared output buffer and dedicated output buffers, and in both cases of a round-robin and a state-dependent plane load balancing schemes. ■

Theorem 8: The minimum output buffer size required is given by the following inequalities.

$$B_{\rm out}(\text{shared}) > (N-K)\,B_r\,(F_{\max}-1) \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,, \tag{22}$$

$$\begin{aligned} B_{\rm out}(\text{dedicated}) = K\,B_{\rm o} \;&>\; K\,Q_{\rm res}^{\rm worst}(\text{dedicated}) = \\ &= K\,(N-K)\,B_r\,(F_{\max}-1) \quad \text{for} \quad m \geq 1 \quad \text{and} \quad s \geq s_{\min} = \frac{1}{K}\,. \end{aligned} \tag{23}$$

Proof: Eqs. (22) and Eq. (23) follow from (21). ■

Remark 3. Note that the above formulas are independent of the number of priorities.

Remark 4. For $N = 64$, $B_r = 32$, $K = 6$, and $F_{\max} = 100$, the output buffer required is of the order of hundred of thousands of packets, which is a prohibitively large size. To avoid this problem, a mechanism should be implemented in order to unblock potentially blocked-lower priority packets.

III. Numerical Results

Here we present the minimum output buffer $B_{\rm out}$ required by high-priority traffic for a safe operation of a 64x64 PPS/buffered-crossbar switch running at an OC-192 link speed. The number of planes range from 4 to 8, i.e. $N = 64$ and $K = 4, \ldots, 8$. Two cases for the speed and the number of ports of the input/output adaptors are considered: either four subports at OC-192 speed ($s = 1$, $m = 4$), or one port at OC-768 speed ($s = 0.25$, $m = 1$).

A. Round-Robin Load Balancing

Table I shows the minimum value of the switch speed, $s_{\rm min}$, for which the analysis holds according to Eqs. (14) and (15).

TABLE I

Minimum switch speed, $s_{\rm min}$.

K	Shared	Dedicated	
	$m \geq 1$	$m = 1$	$m \geq 2$
4	0.250	0.332	0.333
5	0.200	0.249	0.250
6	0.167	0.199	0.200
7	0.143	0.166	0.167
8	0.125	0.142	0.143

TABLE II

Minimum output buffer size required under round-robin load balancing.

K	OC-192		OC-768	
	Shared	Dedicated	Shared	Dedicated
4	6144	10923	6144	$>$ 8192
5	8192	12800	8192	12760
6	10240	14746	10240	14707
7	12288	16725	12288	16688
8	14336	18725	14336	18688

Table II shows the minimum output buffer size required (in number of packets) for a safe operation assuming $B_r = 32$ in the following two cases:

Case 1: Four subports at OC-192

The input/output adaptors contain four subports at OC-192 speed. The results apply in all cases because the switch port speed is equal to one ($s = 1$) which is higher than all of the entries of Table I.

Case 2: One port at OC-768

The input/output adaptors contain one port at OC-768 speed. In this case $m = 1$ and the ratio of switch port speed to port speed is equal to 0.25 ($s = 0.25$). As the switch port speed is equal to 0.25, only the entries of Table I for $K \geq 5$ apply.

The entries of the four columns are evaluated based on Eqs. (14), (15.b), (14), and (15.a), respectively. The first entry of the last column is bounded according to (6).

B. State-Dependent Load Balancing

Table III shows the minimum output buffer size required (in number of packets) for a safe operation in the case of the state-dependent load balancing scheme. The entries of the two columns are evaluated based on Eqs. (19) and (20), respectively. The two cases of OC-192 and OC-768 correspond to $s = 1$ and $s = 0.25$, respectively. As K takes values in the range $4, \ldots, 8$, the $s_{\min}$ takes values in the range $0.25, \ldots, 0.125$. Therefore, the results obtained are valid in both cases because $s \geq s_{\min}$ for all values of K.

TABLE III

MINIMUM OUTPUT BUFFER SIZE REQUIRED UNDER STATE-DEPENDENT LOAD BALANCING.

K	OC-192 or OC-768	
	Shared	Dedicated
4	189	> 252
5	252	315
6	315	378
7	378	441
8	441	504

IV. Conclusions

The resequencing egress buffer requirements of a PPS architecture based on a buffered crossbar switch fabric were investigated. An analytical method was developed for evaluating the worst-case resequence queue size and the corresponding egress buffer size required. Two input load balancing schemes were considered: a static one based on a round-robin scheme and a dynamic one based on a state-dependent scheme. It was found that the state-dependent scheme results in reduced size requirements for the egress buffer. The influence of the output buffer arrangement was also investigated. It was proven that the egress buffer size requirements are reduced when the egress buffer is shared among the planes than when dedicated portions are provided per plane.

APPENDIX A

WORST-CASE RESEQUENCING FOR HIGH-PRIORITY TRAFFIC - SHARED OUTPUT BUFFER

Here we present a sequence of events that lead to the scenario described in Section II-A.1 and the corresponding worst case for the resequence queue size when the output buffer is considered to be shared among the planes.

► **Step 1:**

We start by considering hot spot traffic destined to subport A, i.e. all the traffic arriving at the input adaptors is destined to subport A. As the input rates are greater than the corresponding output rates, queues start building up at the corresponding buffers of the switches as well as at the output buffer B_A corresponding to subport A, as shown in Figure 7.

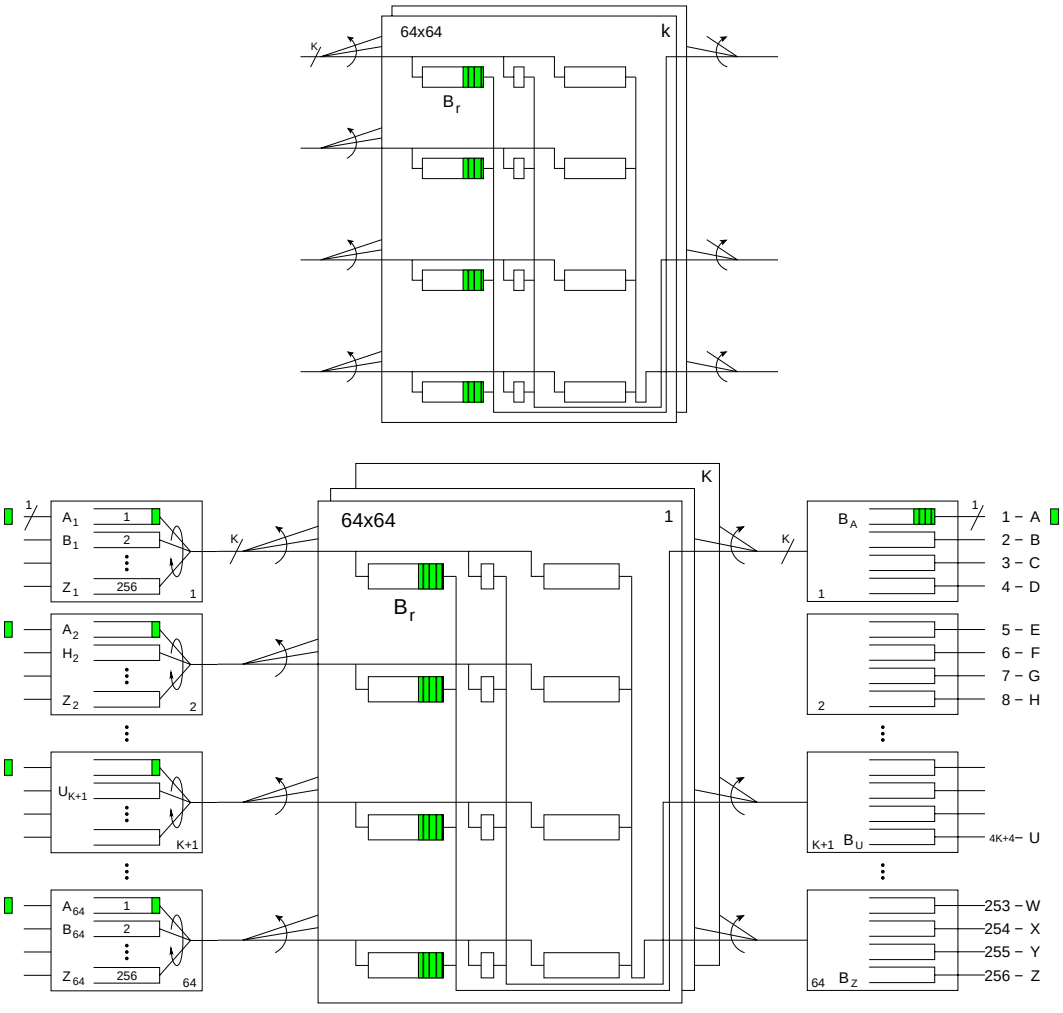

Fig. 7. Step 1: Subport A is a hot spot.

► **Step 2:**

After allowing sufficient time to elapse, the corresponding buffers of the switches as well as the output buffer corresponding to subport A become full as shown in Figure 8. Note that in case that the buffers of the switches are filled before B_A, the input VOQs start also building up. One then arrives at the scenario depicted in Figure 8 by allowing B_A also to fill up and then stopping the incoming traffic until the input VOQs are emptied.

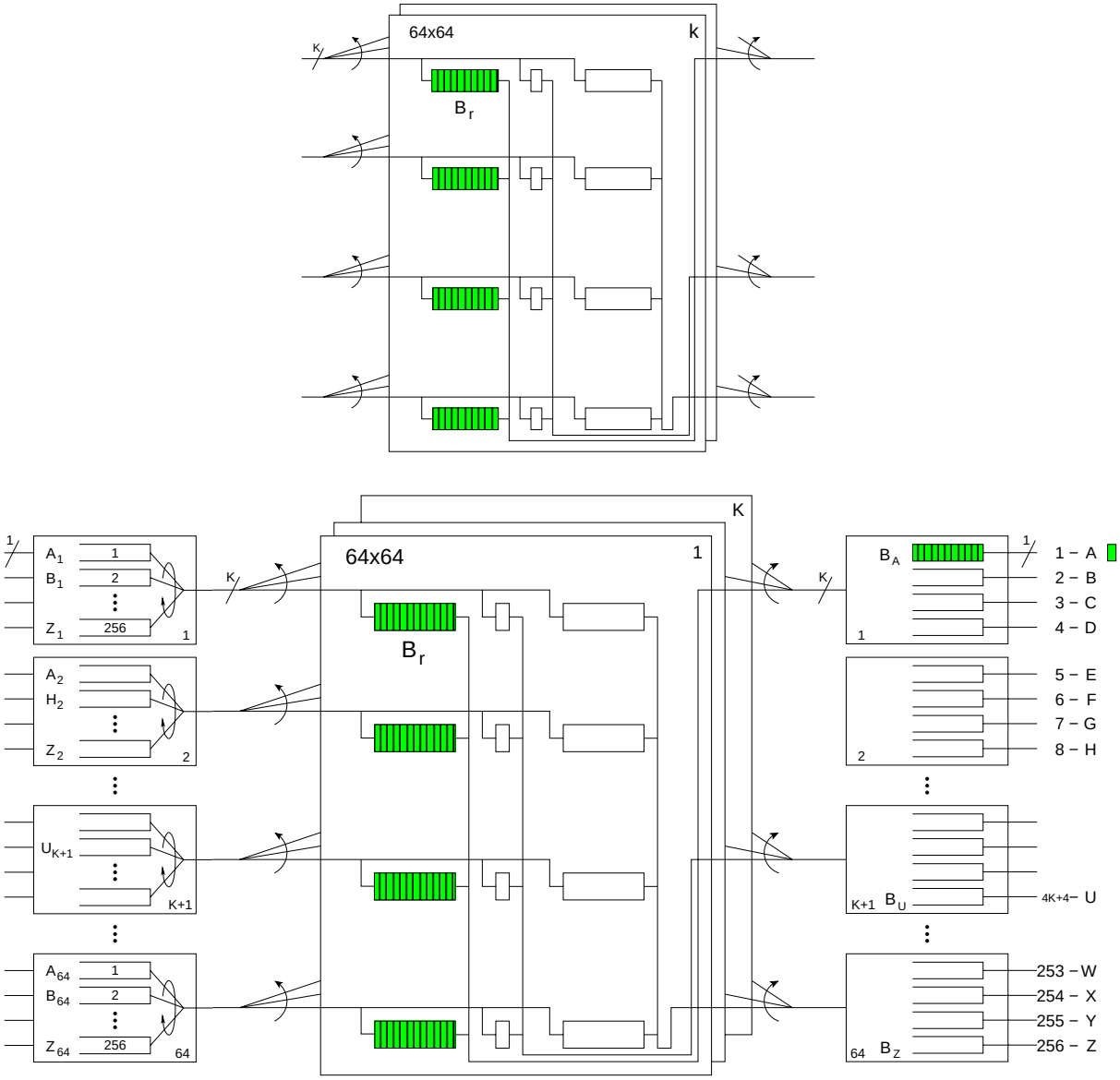

Fig. 8. Step 2: Buffers get full.

► **Step 3:**

The arrival pattern now changes as follows. The first packet (colored blue in Figure 9) of a frame that is destined to subport 1-A arrives at the first input subport of the first input adaptor. Its subsequent packets are assumed to arrive at a rate of one packet every NK time slots. At the same time one packet place becomes available in $B_{1,1}$ of the first plane because of the continuing departures of packets at subport 1-A.

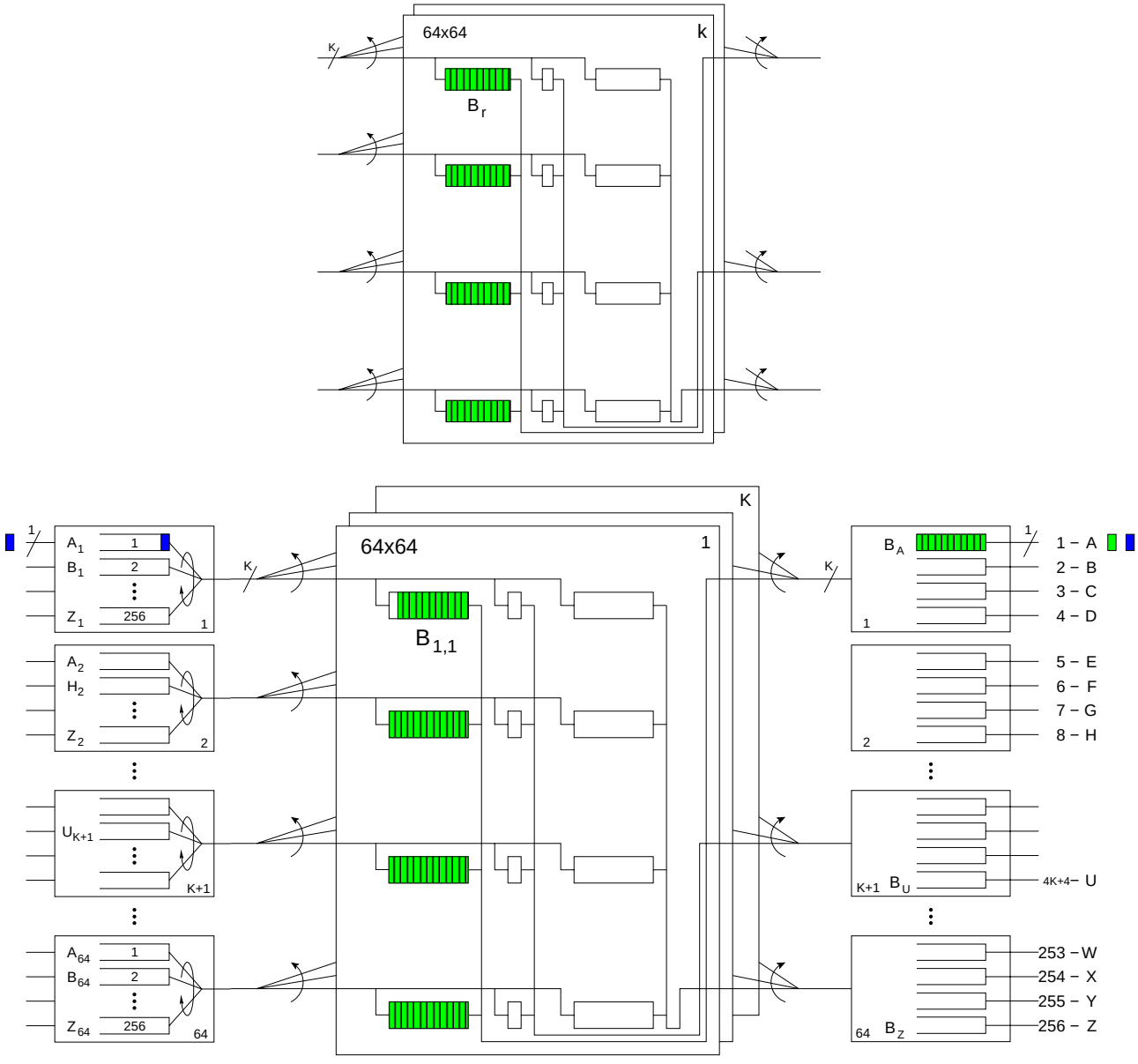

Fig. 9. Step 3: First packet of frame destined to subport A arrives.

► **Step 4:**

The blue packet is transferred to buffer $B_{1,1}$ of the first plane. The first packet (colored brown in Figure 10) of a frame consisting of $K-1$ packets destined to subport 256-Z arrives at the first input subport of the first input adaptor. These packets are assumed to arrive at a rate of one packet every time slot for the next $K-2$ time slots. At the same time one packet place becomes available in $B_{1,1}$ of the second plane because of the continuing departures of packets at subport 1-A.

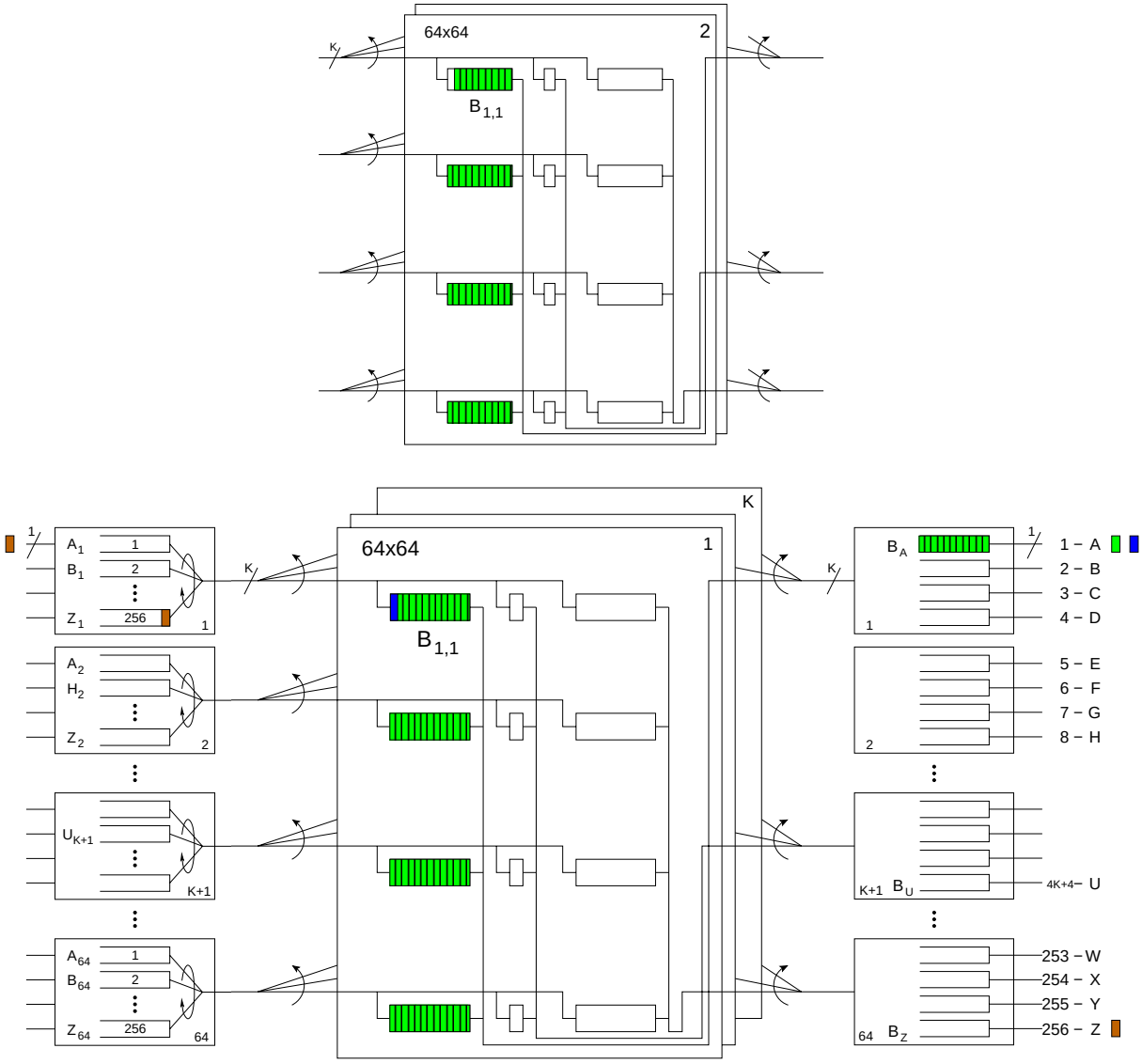

Fig. 10. Step 4: First packet of frame destined to subport Z arrives.

‘

► **Step 5:**

The packets of the frame destined to subport 256-Z are successively transferred to planes 2 through K and subsequently to the corresponding output buffer B_Z, as shown in Figure 11. At the same period there are transfers of green packets from the $B_{1,1}$ buffers of the K planes to B_A because of the continuing departures of packets at subport 1-A.

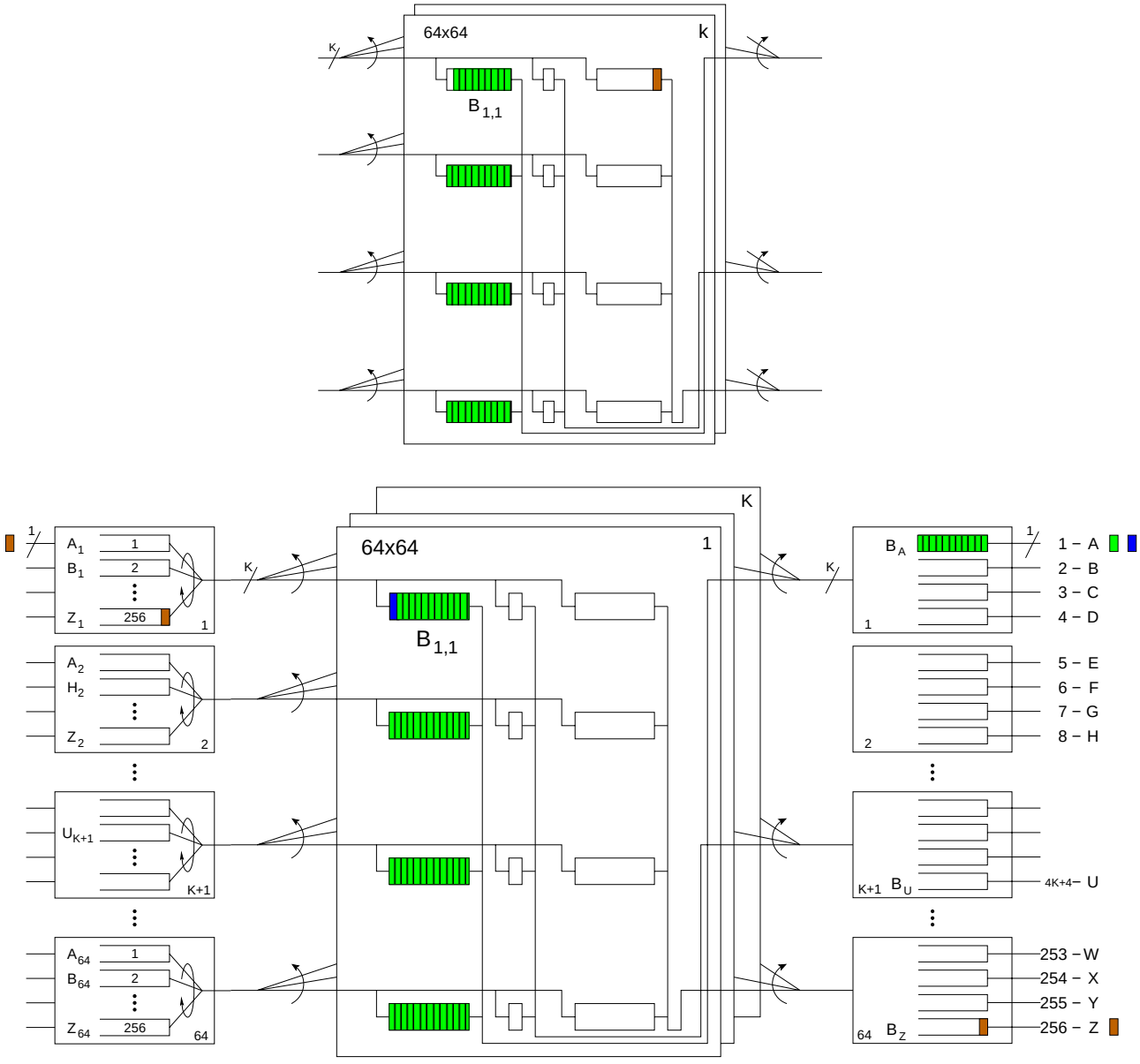

Fig. 11. Step 5: Packets of frame destined to subport Z arrive.

► **Step 6:**

Steps 3 through 5 are now repeated at the second input adaptor, and then successively repeated at input adaptors 3 through 64. Buffers $B_{1,1}, \ldots, B_{64,1}$ of the first plane are full, whereas buffers $B_{1,1}, \ldots, B_{64,1}$ of the remaining planes are no longer full as shown in Figure 12. The arrival pattern described in the previous steps (starting at step 3) is now repeated so that blue packets are always transferred to the first plane, according to the periodicity of the plane load balancing schemes. In each cycle of NK time slots, there is one blue packet arriving and one green packet departing from each of the $B_{i,1}$ ($i = 1, \ldots, 64$) buffers of the first plane. This ensures that these buffers are always full, in contrast to the $B_{i,1}$ ($i = 1, \ldots, 64$) buffers of the remaining planes which are continuously depleted until they become empty, as shown in Figure 13. This also proves that the scenario described in Section II-A.1 is feasible.

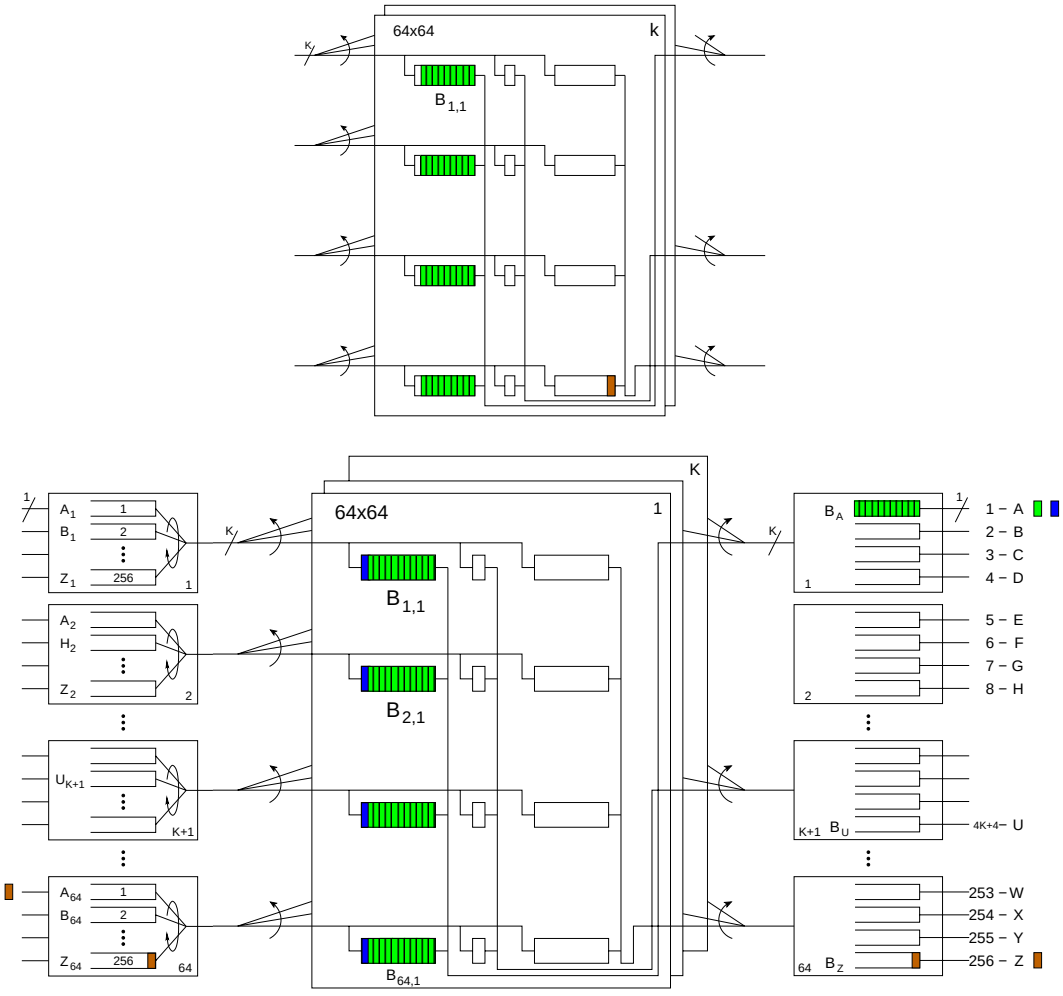

Fig. 12. Step 6: Arrival pattern is repeated.

► **Step 7:**

According to the round-robin scheme employed, the last green packet to be transferred to the first output adaptor is taken from buffer $B_{64,1}$ of plane K as shown in Figure 13. At that instant, the arrival pattern changes as follows. The first packet of a new flow of packets (colored red) destined to subport 1-A arrives at the first input subport of the first input adaptor. The packets of this flow are assumed to arrive at a constant rate of one packet every time slot.

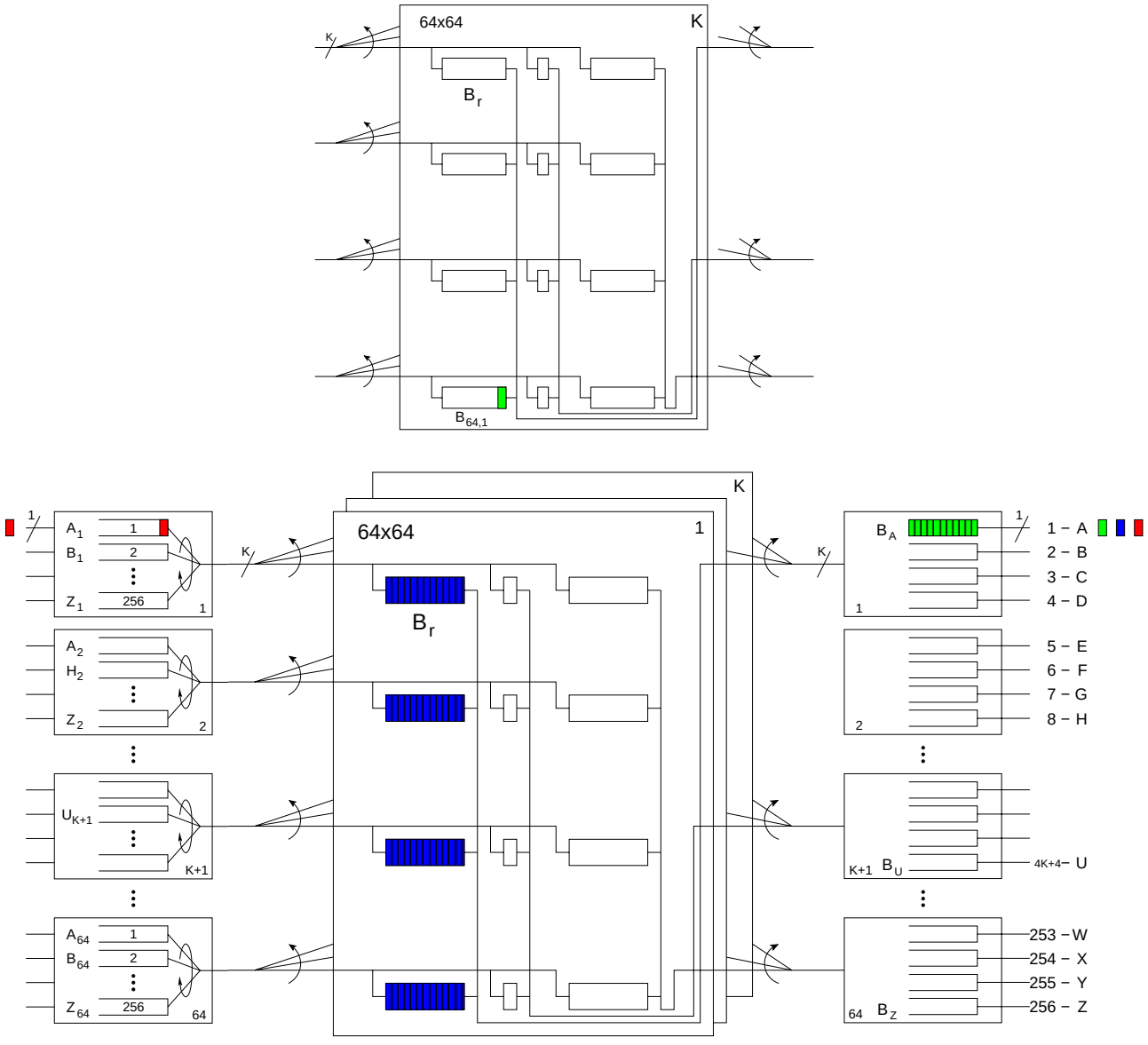

Fig. 13. Step 7: First packet of a frame destined to subport A arrives.

► **Step 8:**

The first blue packet is transferred from buffer $B_{1,1}$ of the first plane to buffer B_A of the corresponding output adaptor, as shown in Figure 14. At the same time the first red packet is transferred from the input VOQ A_1 to the buffer $B_{1,1}$ of the first plane by filling the empty packet place just created. The second red packet is dispatched to the $B_{1,1}$ buffer of the second plane.

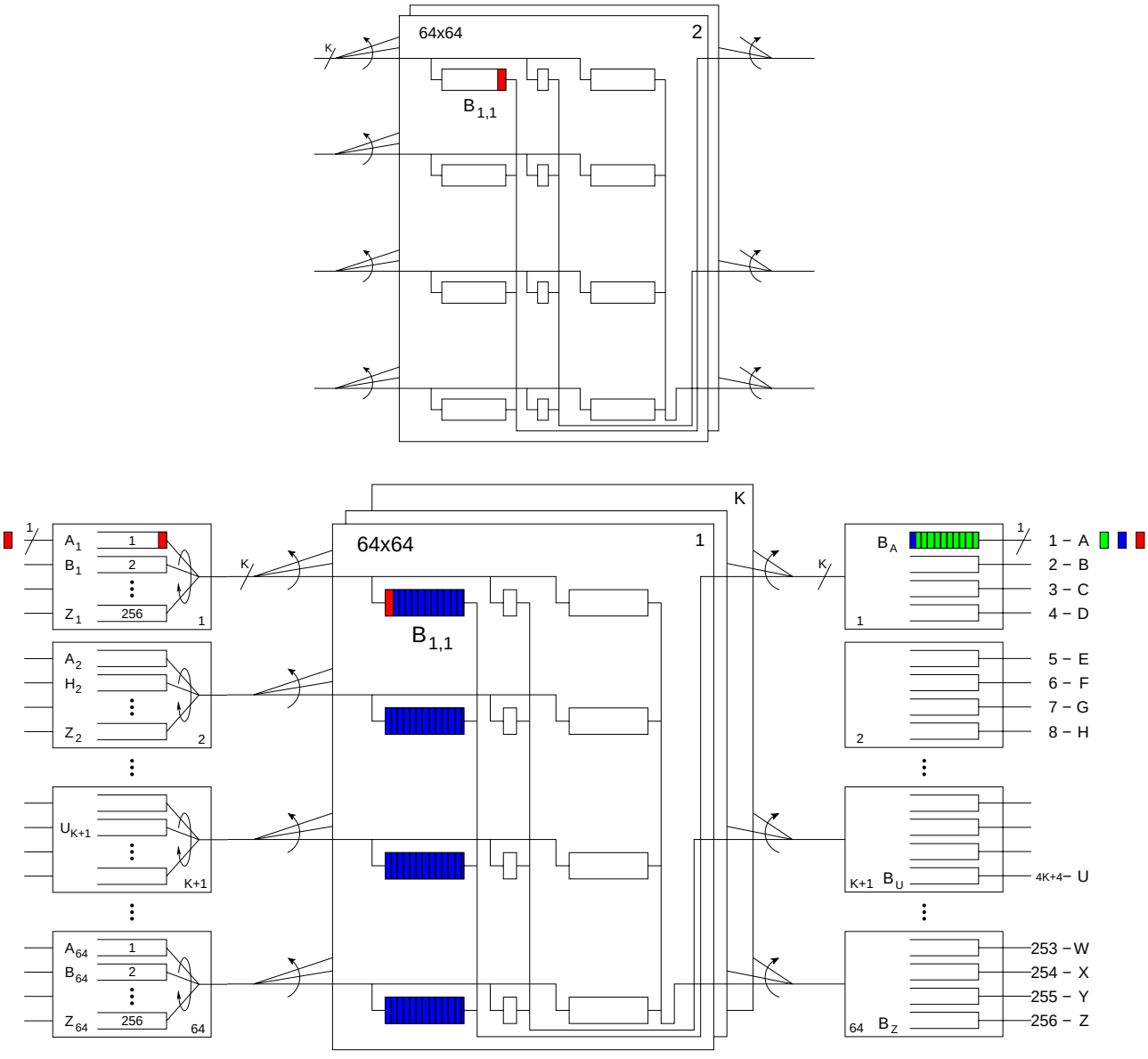

Fig. 14. Step 8: Packets of frame destined to subport A arrive.

► **Step 9:**

The red packets of the frame destined to subport A are successively transferred to planes 2 through K and subsequently to the corresponding output buffer B_A, as shown in Figure 15. Note that in each of the planes 2 through K there is a red packet transferred from the input adaptor every K slots. This, in conjunction with the fact that buffer B_A is full and it is emptied at a rate of one packet per time slot, implies that buffer B_A is fed successively by one packet per time slot, in a round-robin fashion. All the red packets arriving are stored in the resequence queue as they all have to wait for the first red packet stored in buffer $B_{1,1}$ of the first plane.

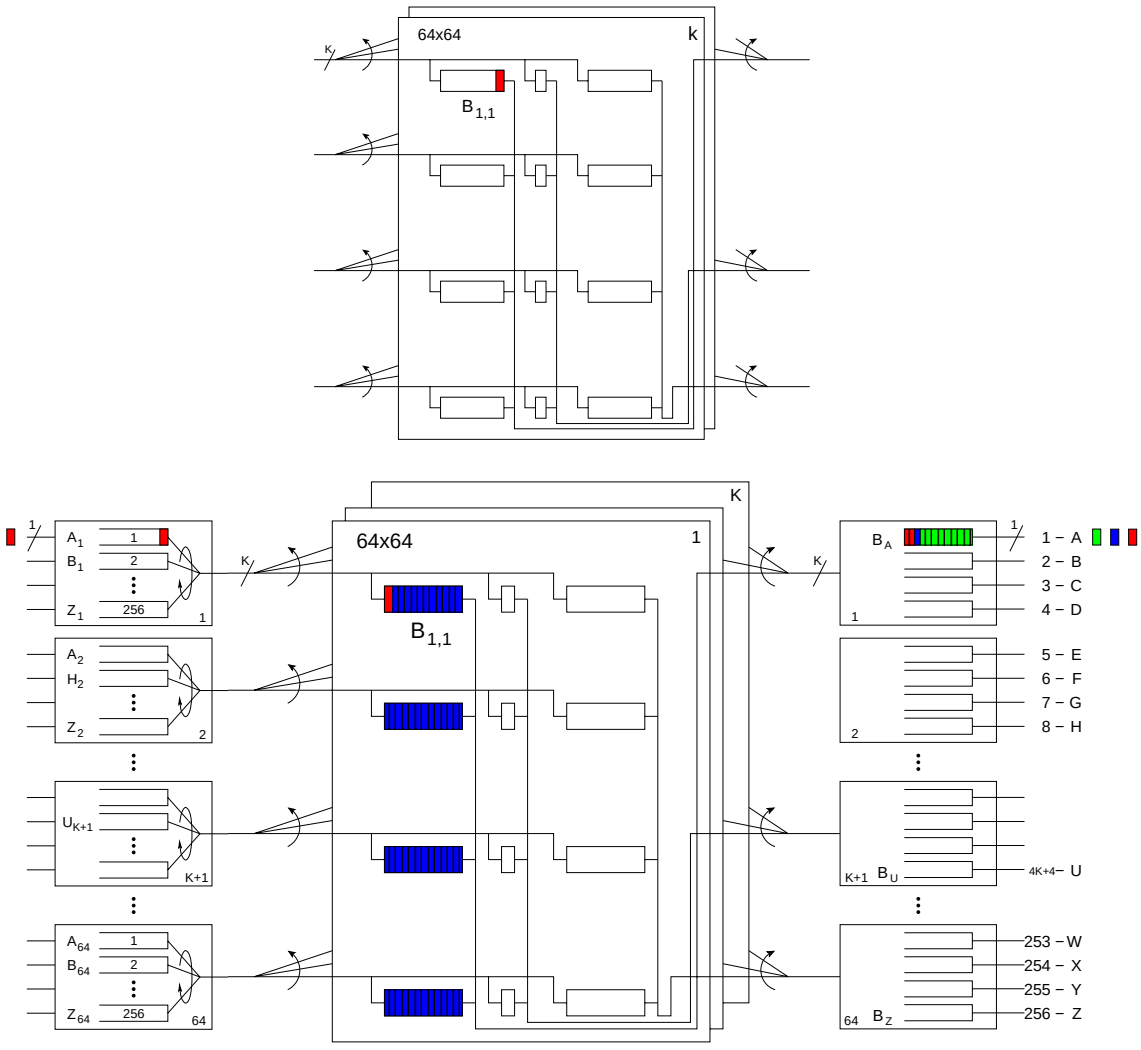

Fig. 15. Step 9: Packets of frame destined to subport A arrive.

► **Step 10:**

The red packet in buffer $B_{1,1}$ of the last plane is transferred to the corresponding output buffer B_A, as shown in Figure 16. Thus, a number of $K - 1$ red packets are already stored in the resequence queue as they all have to wait for the first red packet stored in buffer $B_{1,1}$ of the first plane. Furthermore, the stream of red packets is interrupted by the first packet (colored brown in Figure 16) of a frame arriving at the first input subport of the first input adaptor and destined to subport 256-Z. This packet is therefore placed at VOQ Z_1. The brown packets are assumed to arrive at a rate of one packet every K time slots in a way that they are all transferred to the first plane, whereas all the packets (except the first one) of the red flow are transferred to the remaining planes.

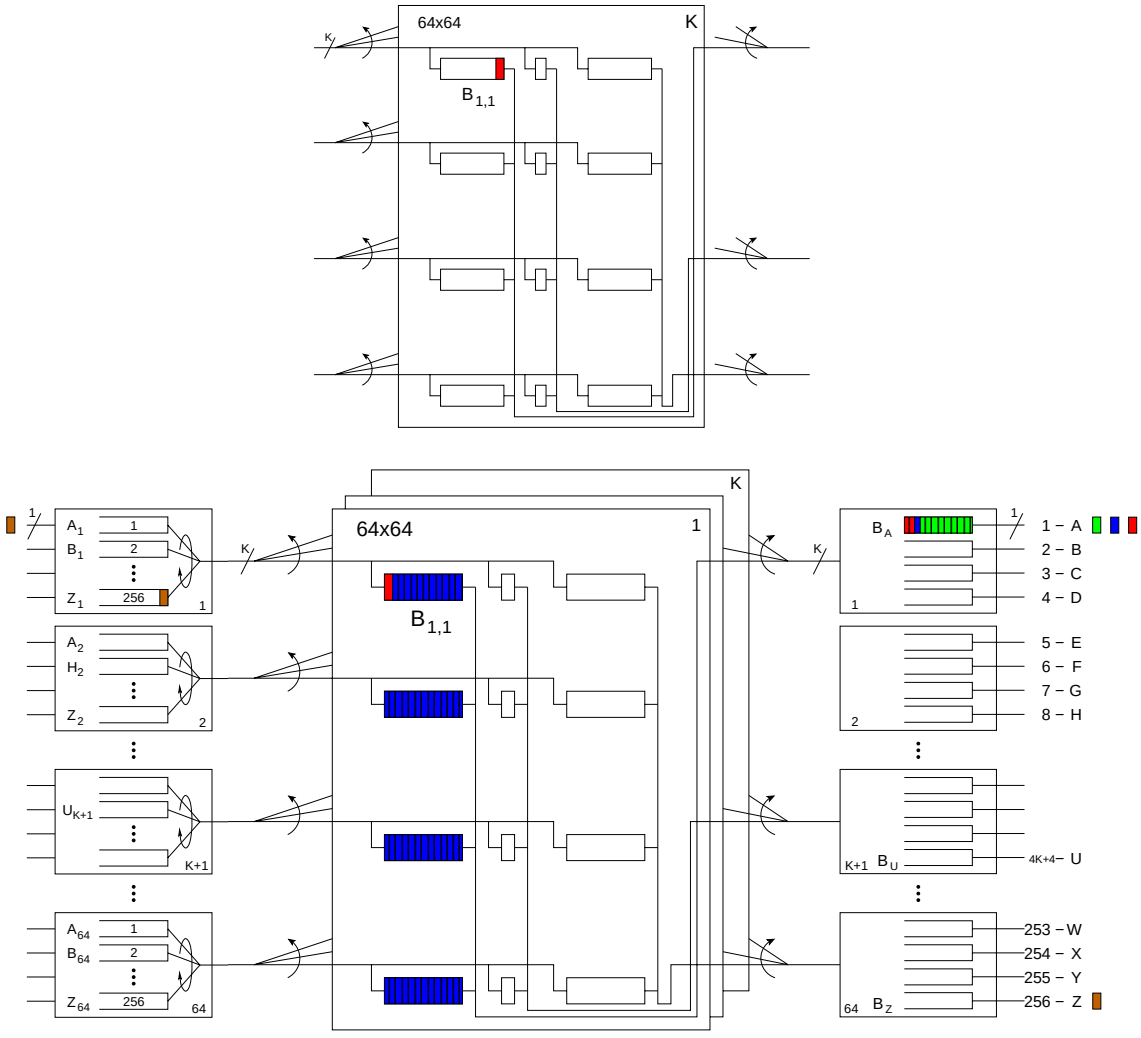

Fig. 16. Step 10: First packet of a frame destined to subport Z arrives.

► **Step 11:**

The first blue packet in $B_{2,1}$ buffer of the first plane is transferred to the corresponding output buffer B_A, as shown in Figure 17. Also the brown packet is dispatched from VOQ Z_1 to buffer $B_{1,64}$ of the first plane and the flow of red packets resumes.

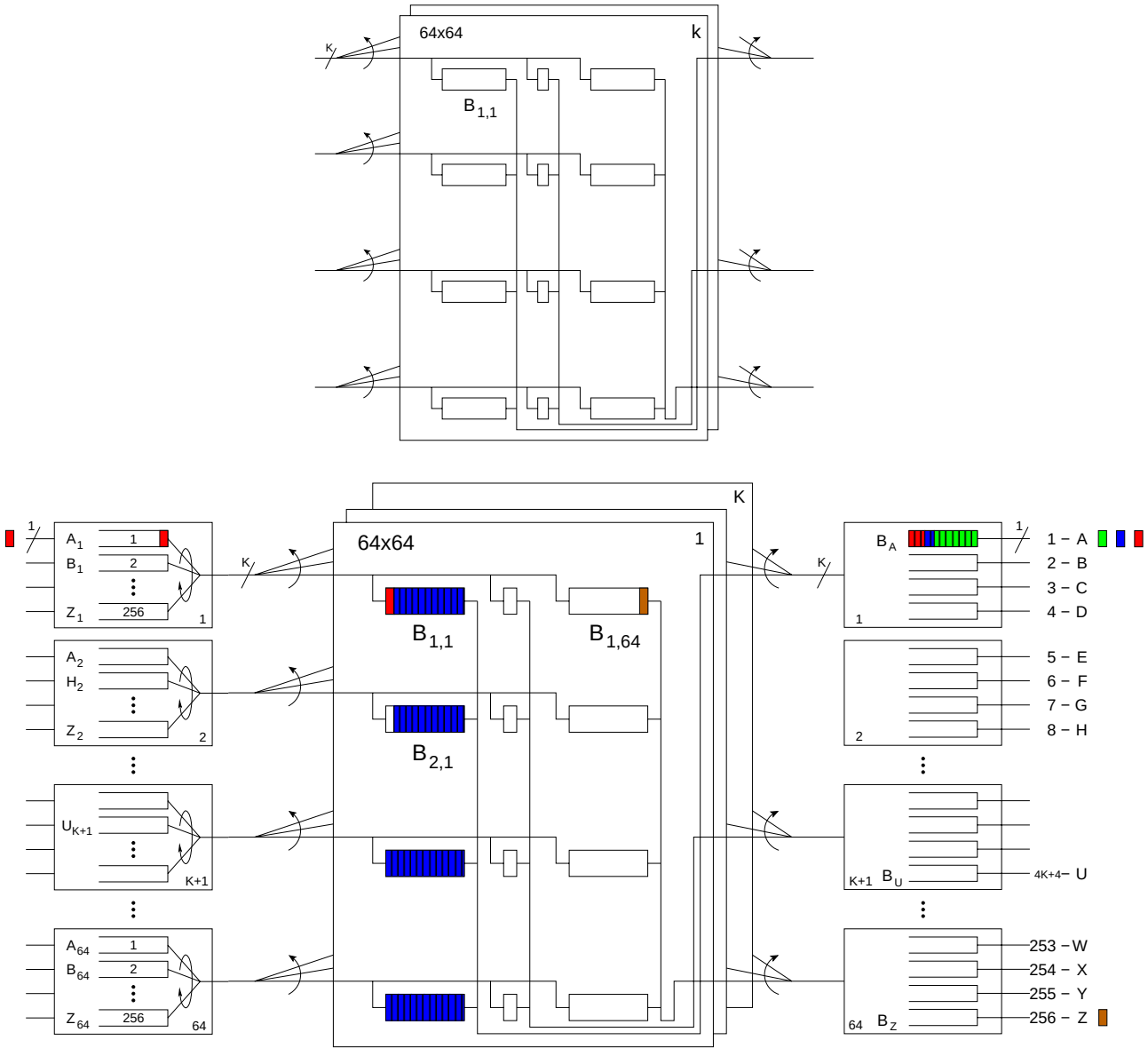

Fig. 17. Step 11: Flow of red packets resumes its arrival.

► **Step 12:**

The brown packet is transferred from buffer $B_{1,64}$ of the first plane to its destination output buffer B_Z, as shown in Figure 18. The red packets of the frame destined to subport A are successively transferred to planes 2 through K and subsequently to the corresponding output buffer B_A.

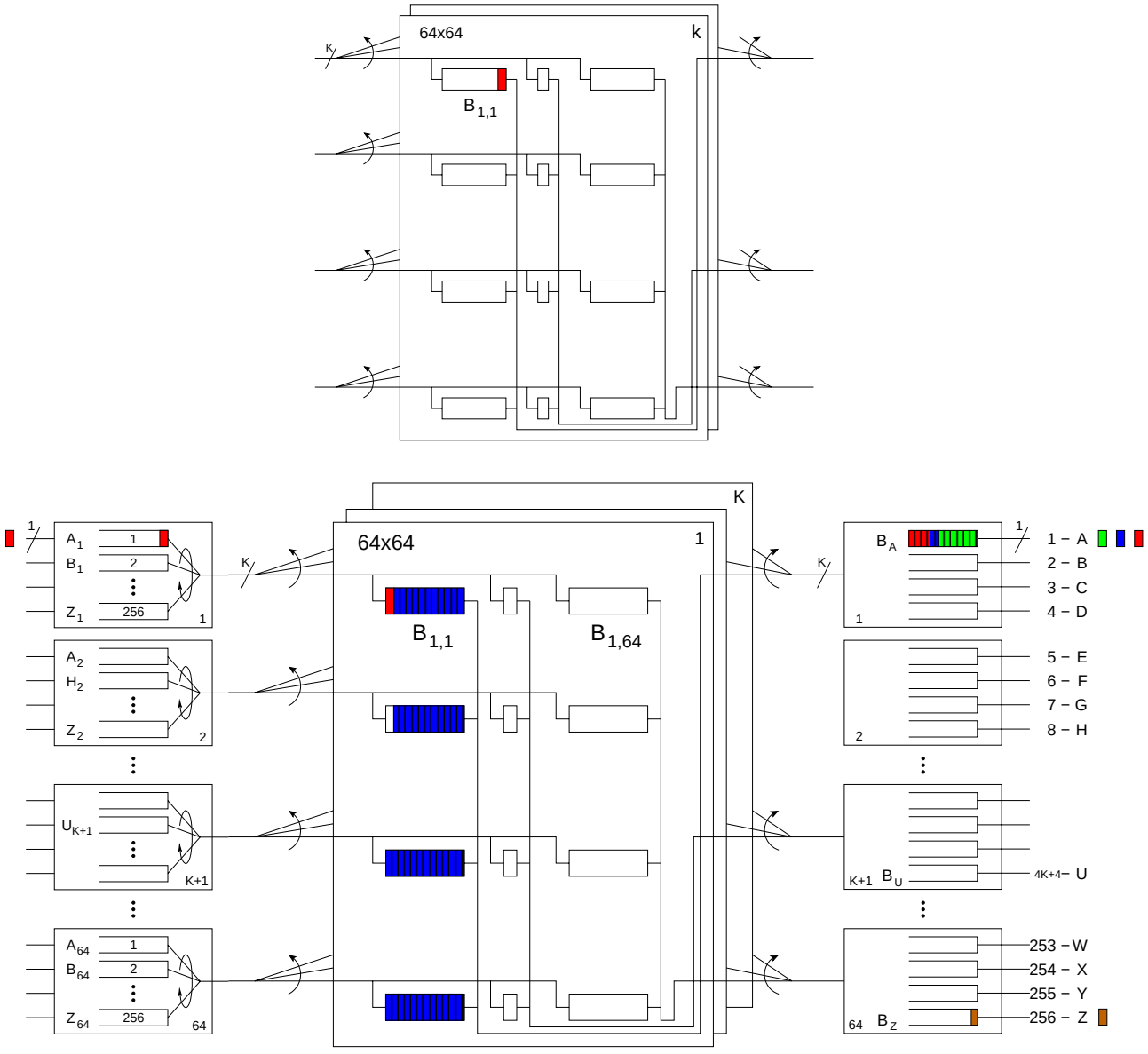

Fig. 18. Step 12: Packets are continuously dispatched to buffer B_A.

► **Step 13:**

The pattern of packet arrivals during time slots 1 through K of the previous cycle is repeated. There is one place empty in buffers $B_{2,1}$ through $B_{64,1}$ of the first plane, as illustrated in in Figure 19.

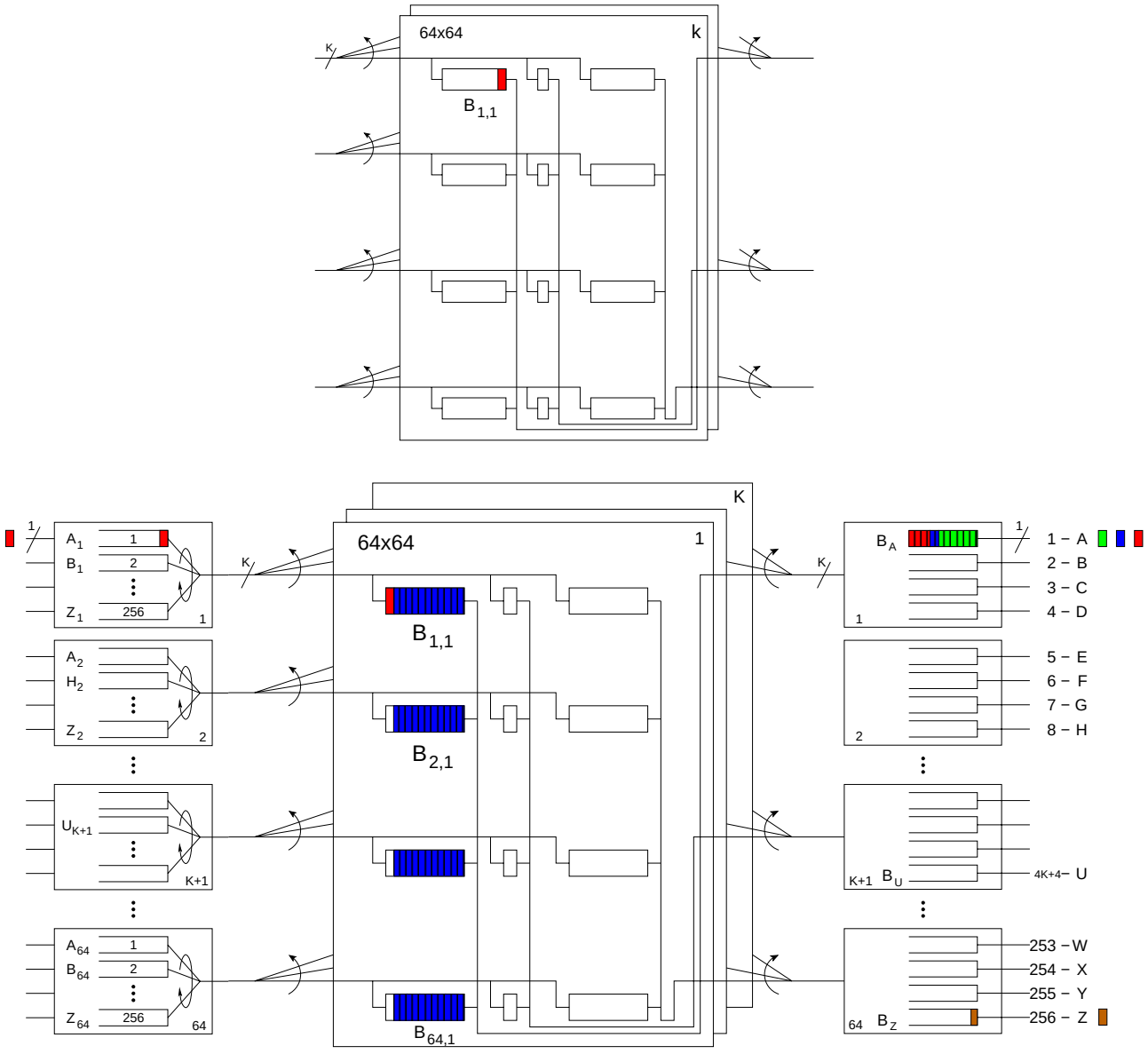

Fig. 19. Step 13: Dispatch pattern to buffer B_A is repeated.

► **Step 14:**

Buffer B_A is always full as it is emptied at a rate of one packet per time slot and is fed by one packet per time slot from the planes. For every blue packet transferred from the first plane, the round-robin scheme results in $K-1$ subsequent red packet transfers from planes 2 through K. These red packets are stored in the resequence queue as they all have to wait for the first arrived red packet stored in buffer $B_{1,1}$ of the first plane. Note that there are always red packets stored in planes 2 through K because for every red packet transferred to buffer B_A there is a corresponding red packet arriving at the input adaptor as shown in Figure 20. Note also that following each departure of a blue packet from plane 1, a brown packet is dispatched from VOQ Z_1 to buffer $B_{1,64}$ of the first plane.

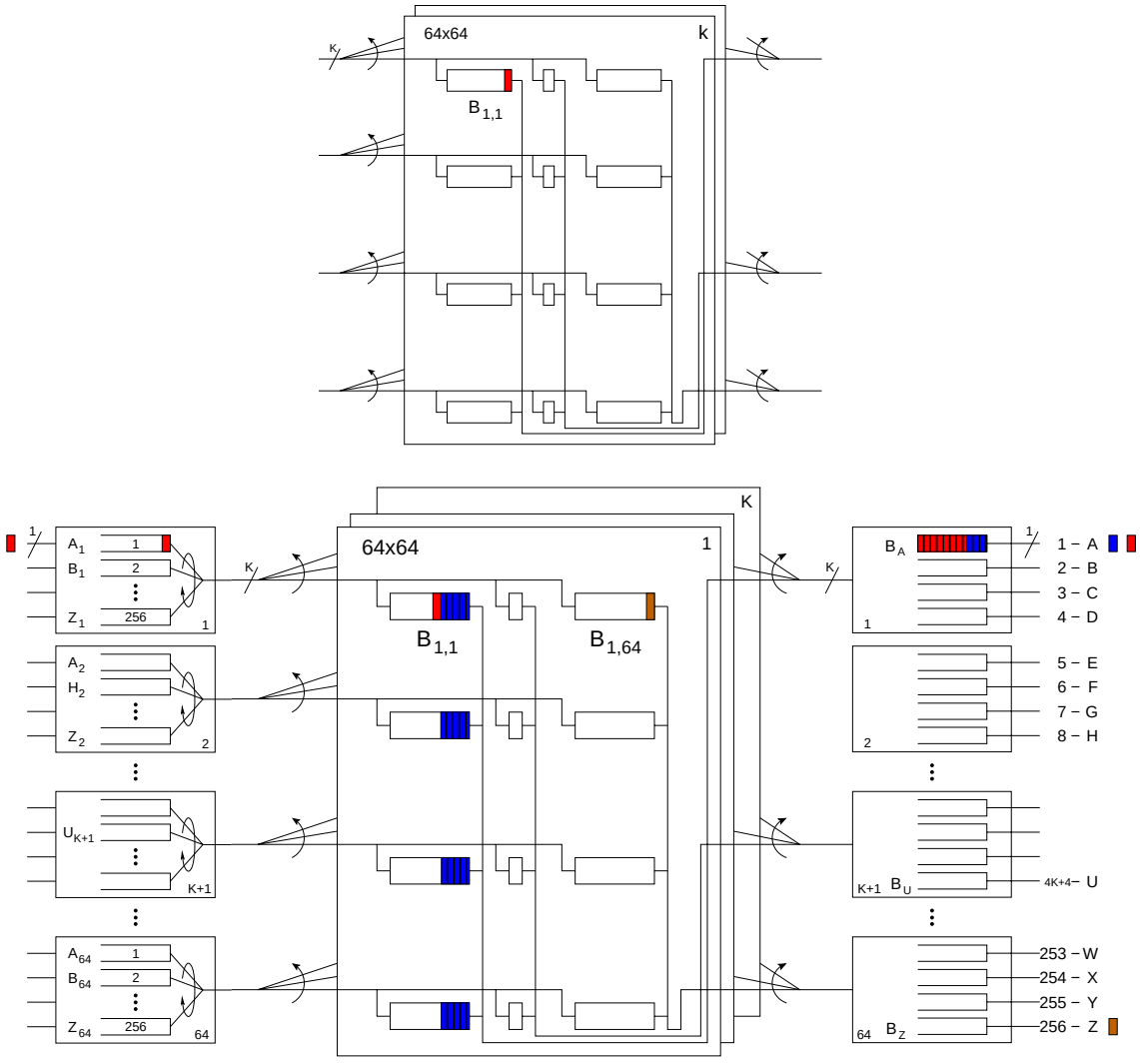

Fig. 20. Step 14: Packets are continuously dispatched to buffer B_A.

► **Step 15:**

The first arrived red packet stored in buffer $B_{1,1}$ of the first plane is transferred to buffer B_A only after all the NB_r blue packets have been transferred from buffers $B_{1,1}, \ldots, B_{64,1}$ of the first plane to buffer B_A, as shown in Figure 21. At that instant the number of red packets stored in the resequence queue is equal to $(K-1)NB_r$ as to every blue packet correspond $K-1$ successive red packet arrivals that need to be resequenced.

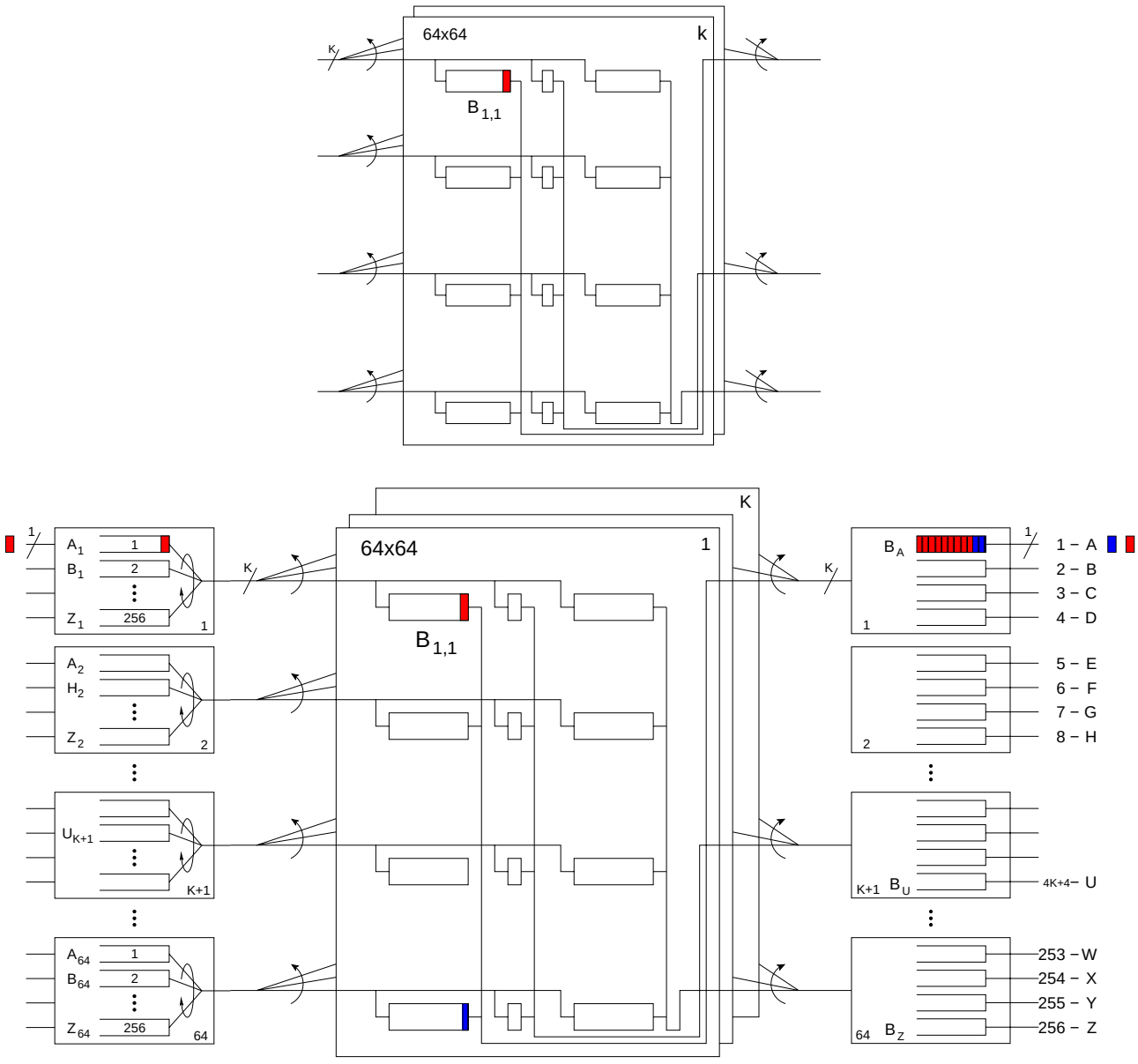

Fig. 21. Step 15: Maximum number of resequenced red packets.

Remark 5. The maximum packet rate observed at any of the switch links for steps 3 through 15 is equal to one packet every K time slots. Moreover, for the queues to build up in Steps 1 and 2, it should hold that $Ks > 1$, or $s > 1/K$. Consequently, the scenario presented holds for any value of the switch port speed s in the range $s > s_{\min} = 1/K$.

Remark 6. The scenario presented obtains the worst case of resequence queue size by considering traffic present on only one subport of any given input or output adaptor. Furthermore, this size cannot be increased by considering more than one subports. Consequently, the number of subports of each adaptor is irrelevant, as it does not affect the outcome.

APPENDIX B
WORST-CASE RESEQUENCING FOR HIGH-PRIORITY TRAFFIC - DEDICATED OUTPUT BUFFERS

Here we present a sequence of events that lead to the worst case for the resequence queue size when each output buffer is assumed to be split to K dedicated buffers, one for each plane.

We start by considering the steps 1 through 6 presented in Appendix A. At the end of step 6, buffers $B_{1,1}, \ldots, B_{64,1}$ of the first plane are full, whereas buffers $B_{1,1}, \ldots, B_{64,1}$ of the remaining planes are empty. Also the dedicated output buffers $B_A(1), \ldots, B_A(K)$, corresponding to planes 1 through K, respectively, are full. We now extend step 6 by maintaining the arrival pattern such that the dedicated output buffers $B_A(2), \ldots, B_A(K)$ start depleting, while output buffer $B_A(1)$ as well as buffers $B_{1,1}, \ldots, B_{64,1}$ of the first plane remain full. Because of the symmetry, the evolution of the buffer occupancy of the dedicated output buffers $B_A(2), \ldots, B_A(K)$ is identical. As it will be seen, the evolution of the buffer, as well as the resequence queue occupancy of these buffers remains identical in the following steps to be presented. Consequently, in the remainder we focus on the representative dedicated output buffer $B_A(k)$ ($k > 1$), as indicated in Figure 22. As the depletion process continues, the buffer occupancy of $B_A(k)$ decreases.

► **Step 7:**

When the buffer occupancy of $B_A(k)$ has dropped to the level G, the arrival pattern changes as follows. The first packet of a new flow of packets (colored red) destined to subport 1-A arrives at the first input subport of the first input adaptor. The packets of this flow are assumed to arrive at a constant rate of one packet every time slot.

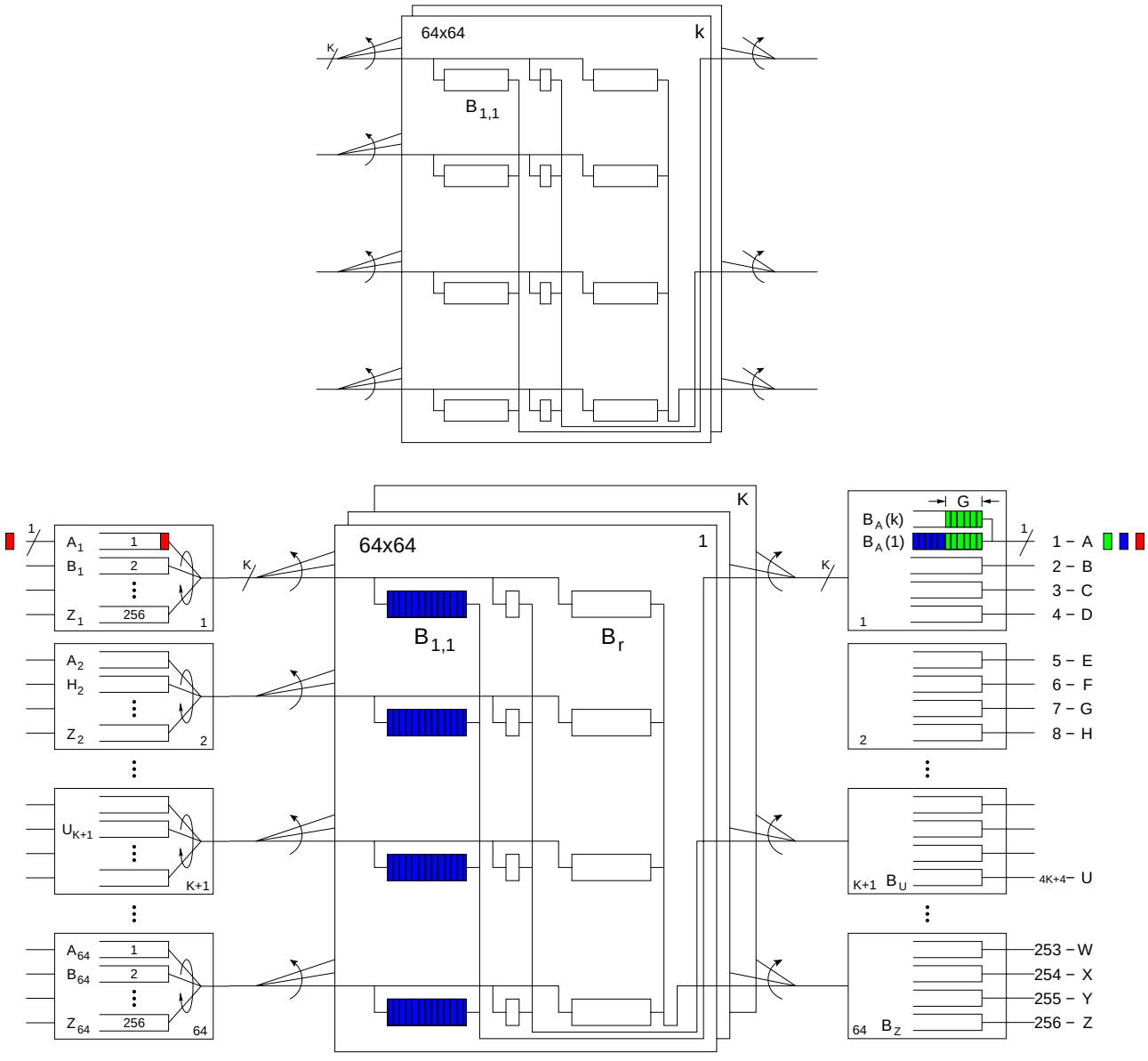

Fig. 22. Step 7: First packet of a frame destined to subport A arrives.

► **Step 8:**

A green packet is transmitted out of buffer $B_A(1)$. The empty place is filled with a blue packet which is transferred from buffer $B_{1,1}$ of the first plane to buffer $B_A(1)$, as shown in Figure 23. At the same time the first red packet is transferred from the input VOQ A_1 to the buffer $B_{1,1}$ of the first plane by filling the empty packet place just created. The second red packet is dispatched to the $B_{1,1}$ buffer of the second plane.

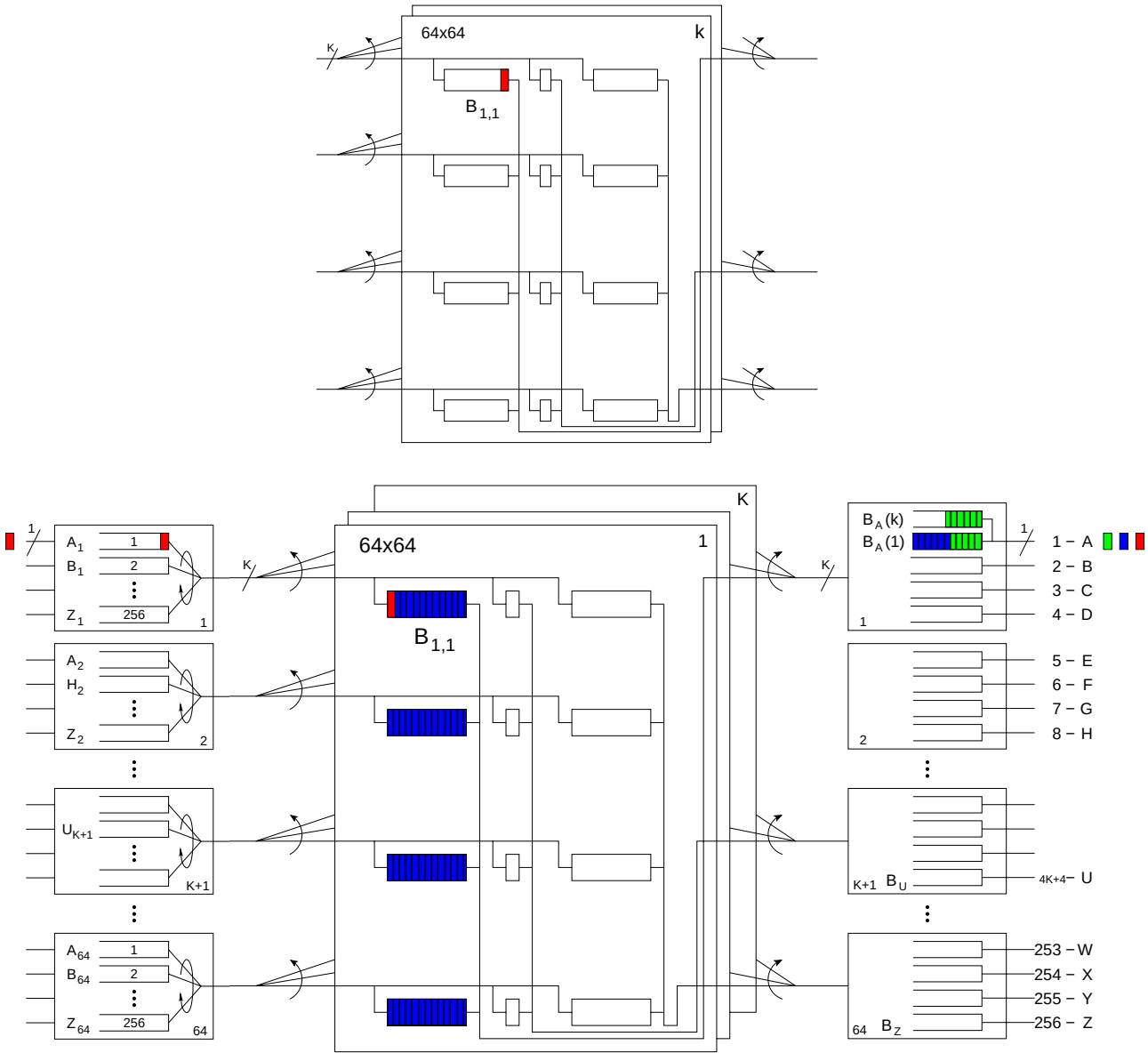

Fig. 23. Step 8: Packets of frame destined to subport A arrive.

As at each time slot one packet is transmitted to the output subport taken from the output buffers in a round-robin fashion, each of the buffers $B_A(i)$ ($i = 1, \ldots, K$) is depleted at a rate of $1/K$ packets per slot for at least a period of KG time units. This implies that plane 1 dispatches one packet to the output buffer $B_A(1)$ every K time slots, which in turn implies that buffer $B_{1,1}$ of the first plane dispatches one packet to the output buffer $B_A(1)$ every NK time slots. According to (9) and (10), the amount of time that the red packet will spend in plane 1 is equal to KNB_r provided that $G \geq NB_r$. Note that during this period the arrival rate of red packets attempting to enter buffer $B_{1,1}$ of the first plane is $1/K$, The fact that this rate is greater than buffer's depletion rate of $1/NK$, implies that this buffer is always full and that the effective arrival rate of red packets to this buffer is equal to $1/NK$ packets per slot. Consequently, the arrival rate of red packets to buffer $B_{1,1}$ of any of the remaining planes is equal to $[1 - (1/NK)]/(K - 1)$.

► **Step 9:**

The red packets of the frame destined to subport A are successively transferred to planes 2 through K and subsequently to the corresponding output buffers $B_A(2), \ldots, B_A(K)$, as shown in Figure 24. All these red packets are stored in the resequence queue as they all have to wait for the first red packet stored in buffer $B_{1,1}$ of the first plane. Note that as long as these buffers are not full, or equivalently buffer $B_A(k)$ is not full, the arrival rate $r_{\max}$ of red cells in $B_A(k)$ is given by

$$r_{\max} = \frac{1 - \frac{1}{NK}}{K - 1} = \frac{1}{K}\left[1 + \frac{N - 1}{N(K - 1)}\right] . \tag{24}$$

Note that this rate exceeds the departure rate of green cells, because it holds that

$$r_{\max} = \frac{1}{K}\left[1 + \frac{N-1}{N(K-1)}\right] > \frac{1}{K}\,, \quad \forall\, K \geq 2\,,\ \forall\, N \geq 2\,. \tag{25}$$

Therefore, the buffer occupancy of $B_A(k)$ is constantly increasing.

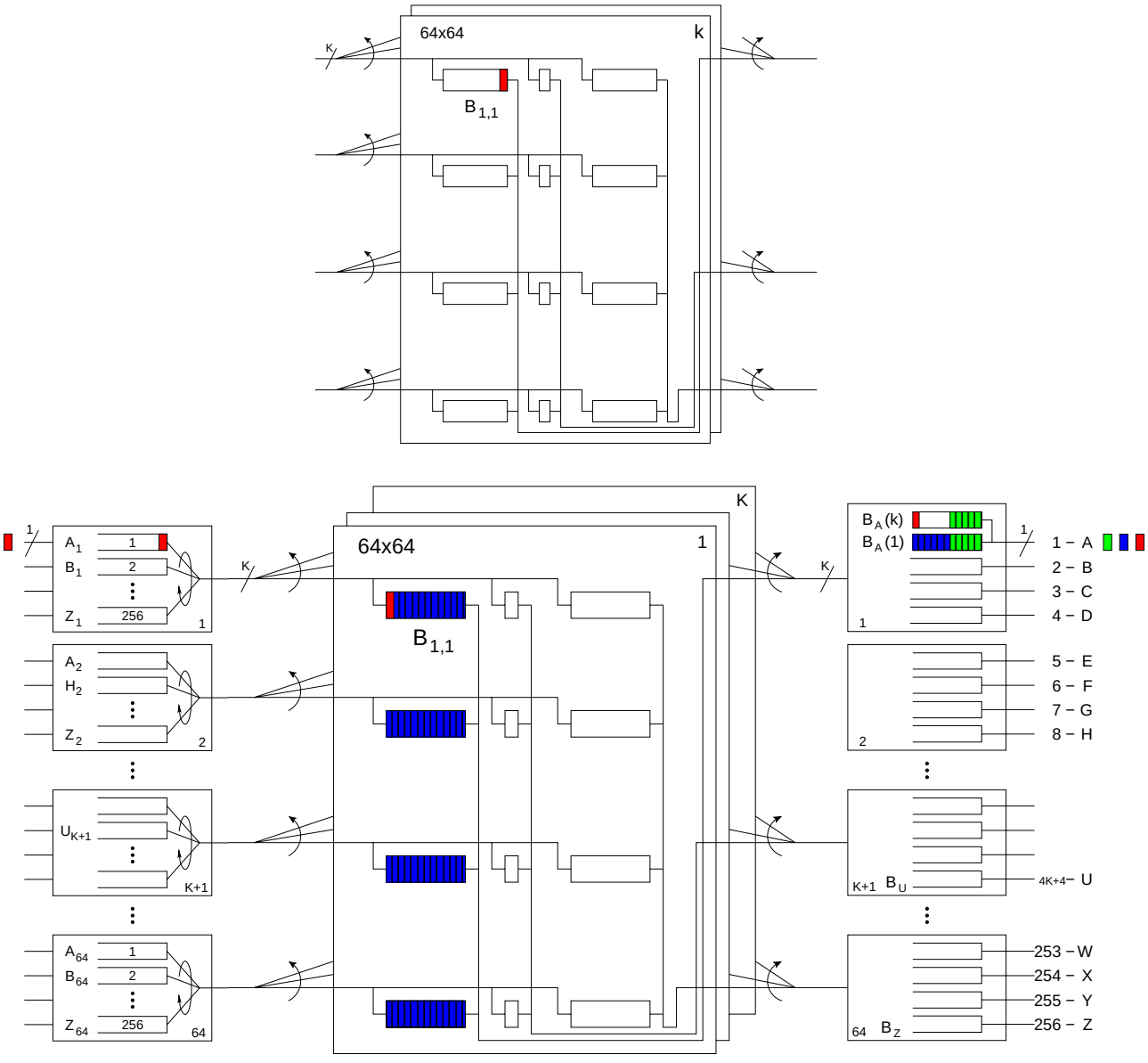

Fig. 24. Step 9: Red packets arrive to output buffer B_A.

► **Step 10:**

The blue packets are transferred from buffers $B_{1,1}, \ldots, B_{64,1}$ of the first plane to the corresponding output buffer $B_A(1)$, as shown in Figure 17. The red packets of the frame destined to subport A are successively transferred to planes 1 through K. In particular, those located in planes 2 through K are subsequently transferred to the corresponding output buffers $B_A(2), \ldots, B_A(K)$.

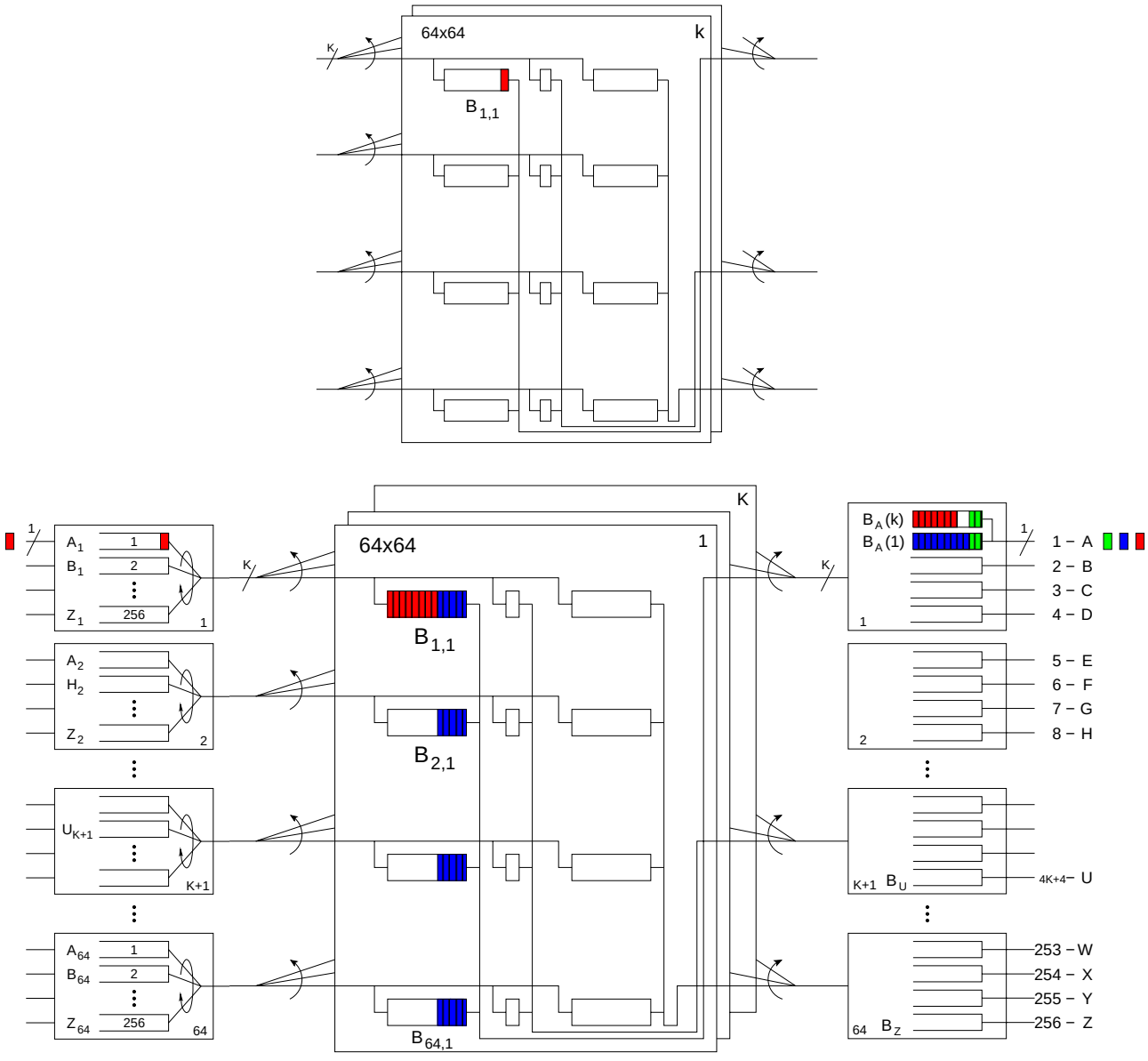

Fig. 25. Step 10: Packets are continuously dispatched to buffer B_A.

► **Step 11:**

The first arrived red packet stored in buffer $B_{1,1}$ of the first plane is transferred to buffer $B_A(1)$ only after all the NB_r blue packets have been transferred from buffers $B_{1,1}, \ldots, B_{64,1}$ of the first plane to buffer $B_A(1)$, as shown in Figure 26. At that instant the number of red packets stored in the resequence queue $B_A(k)$ is given by (2) the product of the arrival rate and the time elapsed. Substituting (9) and (24) into (2) yields

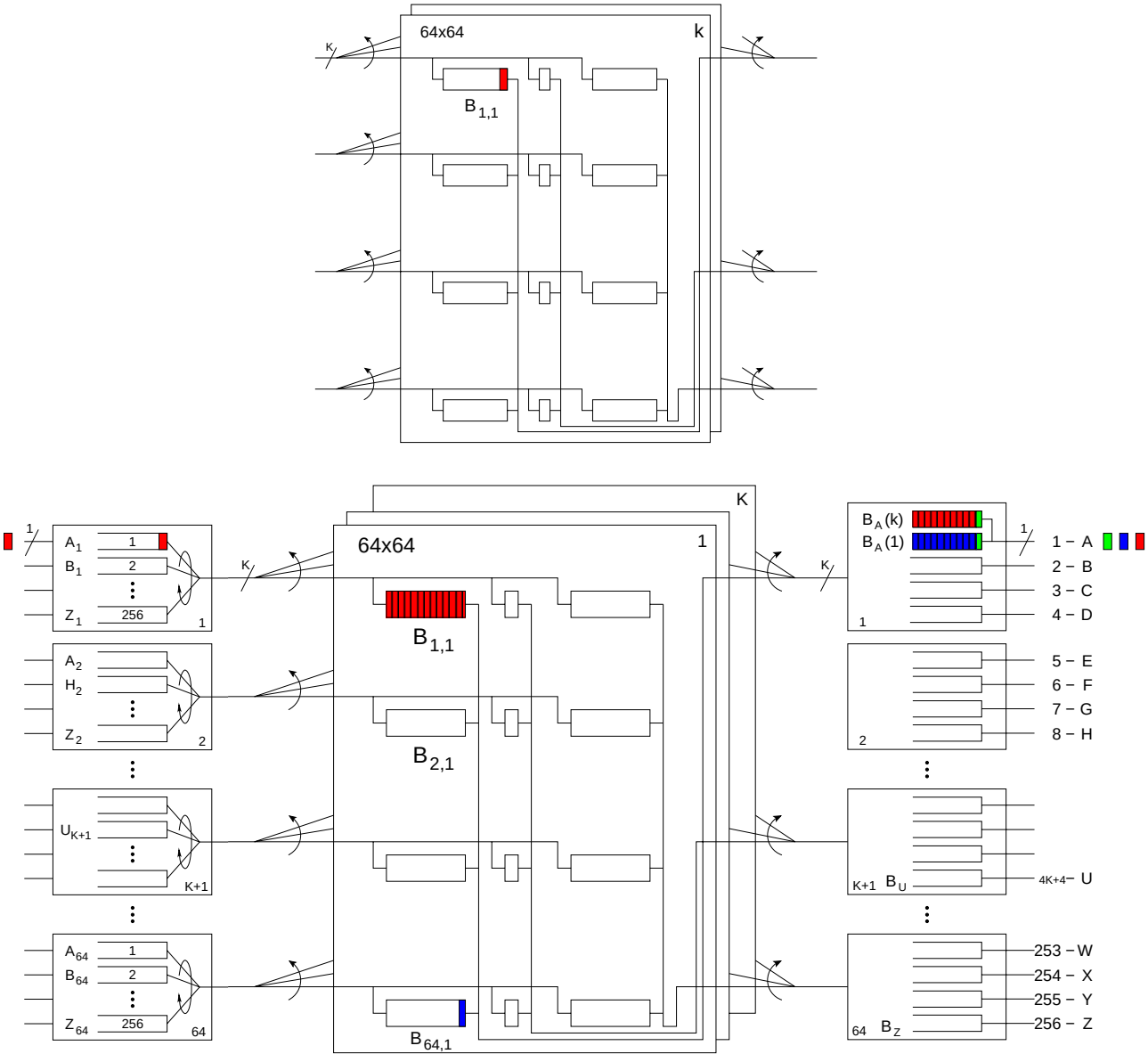

Fig. 26. Step 11: Maximum number of resequenced red packets.

$$Q_{\rm res}^{\rm worst}(\text{dedicated}) = \frac{(KN-1)B_r}{K-1} \quad \text{packets per dedicated output buffer,} \tag{26}$$

such that the total worst-case resequence queue size is given by

$$Q_{\rm res}^{\rm worst}(\text{total dedicated}) = (K-1)\, Q_{\rm res}^{\rm worst}(\text{dedicated}) = (KN-1)B_r\,. \tag{27}$$

In order to ensure a safe operation, the size of the dedicated output buffer $B_{\rm o}$ should exceed the size given by (26). Consequently, the total size of the output buffer should satisfy the following inequality,

$$B_{\rm out}(\text{dedicated}) = K\, B_{\rm o} > K\, Q_{\rm res}^{\rm worst}(\text{dedicated}) = \frac{K}{K-1}\,(KN-1)B_r\,. \tag{28}$$

Note that the above analysis was carried out on the condition expressed by (10), i.e. $G \geq NB_r$. Here we verify that this condition can indeed be satisfied because $B_{\rm o} > [(KN-1)/(K-1)]B_r > NB_r$.

Remark 7. From Eqs. (24) and (25), it follows that the maximum packet rate observed at any of the switch links for steps 3 through 11 is equal to that specified by equation (24). Moreover, for the queues to build up in Steps 1 and 2, it should hold that $Ks > 1$, or $s > 1/K$. Consequently, the scenario presented holds for any value of the switch port speed s is in the range $s \geq s_{\rm min}$, where $s_{\rm min} = r_{\rm max} = \frac{1}{K}\left[1 + \frac{N-1}{N(K-1)}\right]$.

Remark 8. The scenario presented obtains the worst case of resequence queue size by considering traffic present on only one subport of any given input or output adaptor. However, the worst case of resequence queue size can be increased by considering traffic present on the first two subports of the first input adaptor as presented below.

At step 7, let us now consider a new flow of packets (colored brown) destined to subport 1-A arriving at the second input subport of the first input adaptor. Let us also assume that the brown packets arrive at the instants where the round-robin plane load balancing scheme is about to dispatch a packet from VOQ A1 to buffer $B_{1,1}$ of the first plane. Let us further assume that the brown packets are always dispatched to the first plane whereas the red packets arriving simultaneously are dispatched to the second plane. Consequently, all the red packets are dispatched to planes 2 through K implying that the arrival rate $r_{\rm max}$ of red cells in $B_A(k)$ is given by

$$r_{\rm max} = \frac{1}{K-1}\,. \tag{29}$$

Substituting (9) and (29) into (2) yields

$$Q_{\mathrm{res}}^{\mathrm{worst}}(\text{dedicated}) = \frac{K}{K-1} NB_r \quad \text{packets per dedicated output buffer,} \tag{30}$$

such that the total worst-case resequence queue size is given by

$$Q_{\mathrm{res}}^{\mathrm{worst}}(\text{total dedicated}) = (K-1)\, Q_{\mathrm{res}}^{\mathrm{worst}}(\text{dedicated}) = KNB_r\,. \tag{31}$$

In order to ensure a safe operation, the size of the dedicated output buffer B_{o} should exceed the size given by (30). Consequently, the total size of the output buffer should satisfy the following

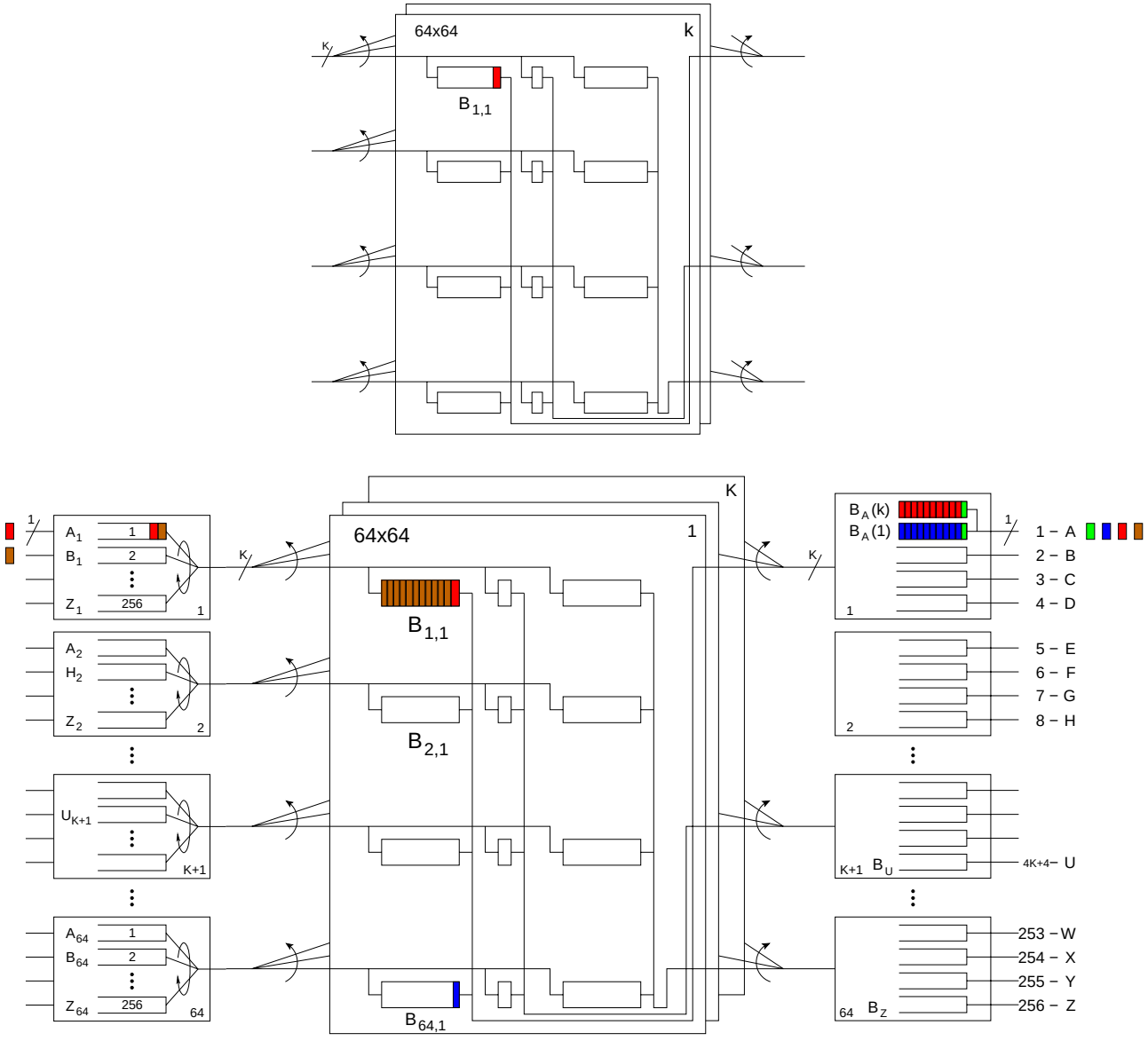

Fig. 27. Step 11: Maximum number of resequenced red packets.

inequality,

$$B_{\text{out}}\,(\text{dedicated}) = K\,B_{\text{o}} > K\,Q_{\text{res}}^{\text{worst}}\,(\text{dedicated}) = \frac{K^2}{K-1}\,NB_r\,. \tag{32}$$

Remark 9. The maximum packet rate observed at any of the switch links in the scenario presented is given by (29). Consequently, the scenario presented holds for any value of the switch port speed s is in the range $s \geq s_{\min}$, where $s_{\min} = r_{\max} = 1/(K-1)$.

APPENDIX C
A STATE-DEPENDENT LOAD BALANCING SCHEME

Here we present a sequence of events that lead to the scenario described in Section II-A.2 and the corresponding worst case for the resequence queue size when the output buffer is considered to be shared among the planes.

► **Step 1:**

We start by considering hot spot traffic destined to subport A as depicted in Step 2 in Appendix A. By stopping the input traffic at the inputs, and after allowing sufficient time to elapse, one arrives at the state depicted in Figure 28. The arrival pattern now changes as follows. The first packet (colored blue in Figure 28) of a frame that is destined to subport 1-A arrives at the first input subport of the first input adaptor. Its subsequent packets are assumed to arrive at a rate of one packet every time slot.

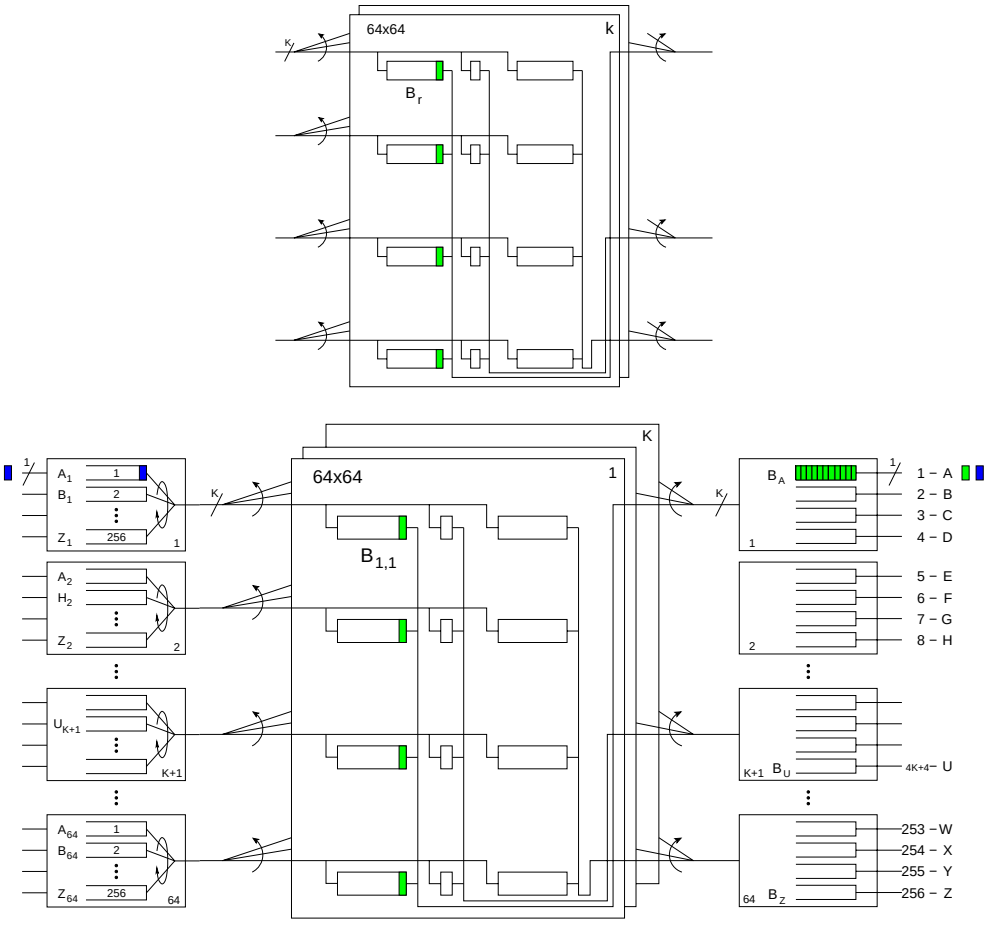

Fig. 28. Step 1: Subport A is a hot spot.

► **Step 2:**

At the same time one packet place becomes available in $B_{1,1}$ of the first plane because of the continuing departures of packets at subport 1-A and corresponding transfers of green packets to output buffer B_A. The blue packet is transferred to buffer $B_{1,1}$ of the first plane. The packet places becoming subsequently available in buffers $B_{1,1}$ of the remaining planes are filled with blue packets as shown in Figure 29.

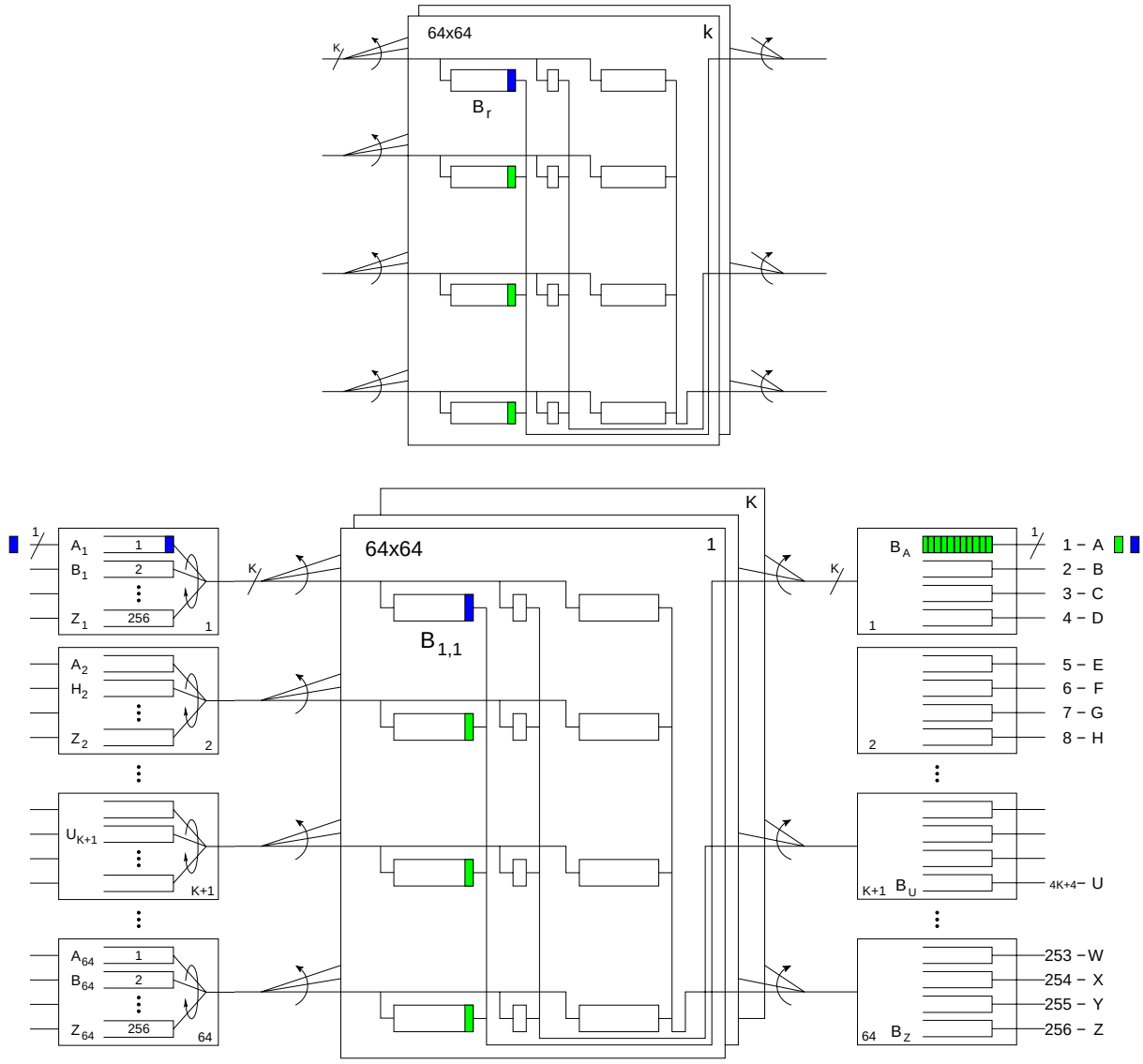

Fig. 29. Step 2: Packet of frame destined to subport A arrive.

‘

► **Step 3:**

Green packets are successively transferred from buffers $B_{2,1}$ of planes 1 through K to output buffer B_A in a round-robin fashion followed by the packets of buffers $B_{3,1}, \ldots, B_{64,1}$. At the same time blue packets are accumulated in buffers $B_{1,1}$ of the various planes as shown in Figure 30.

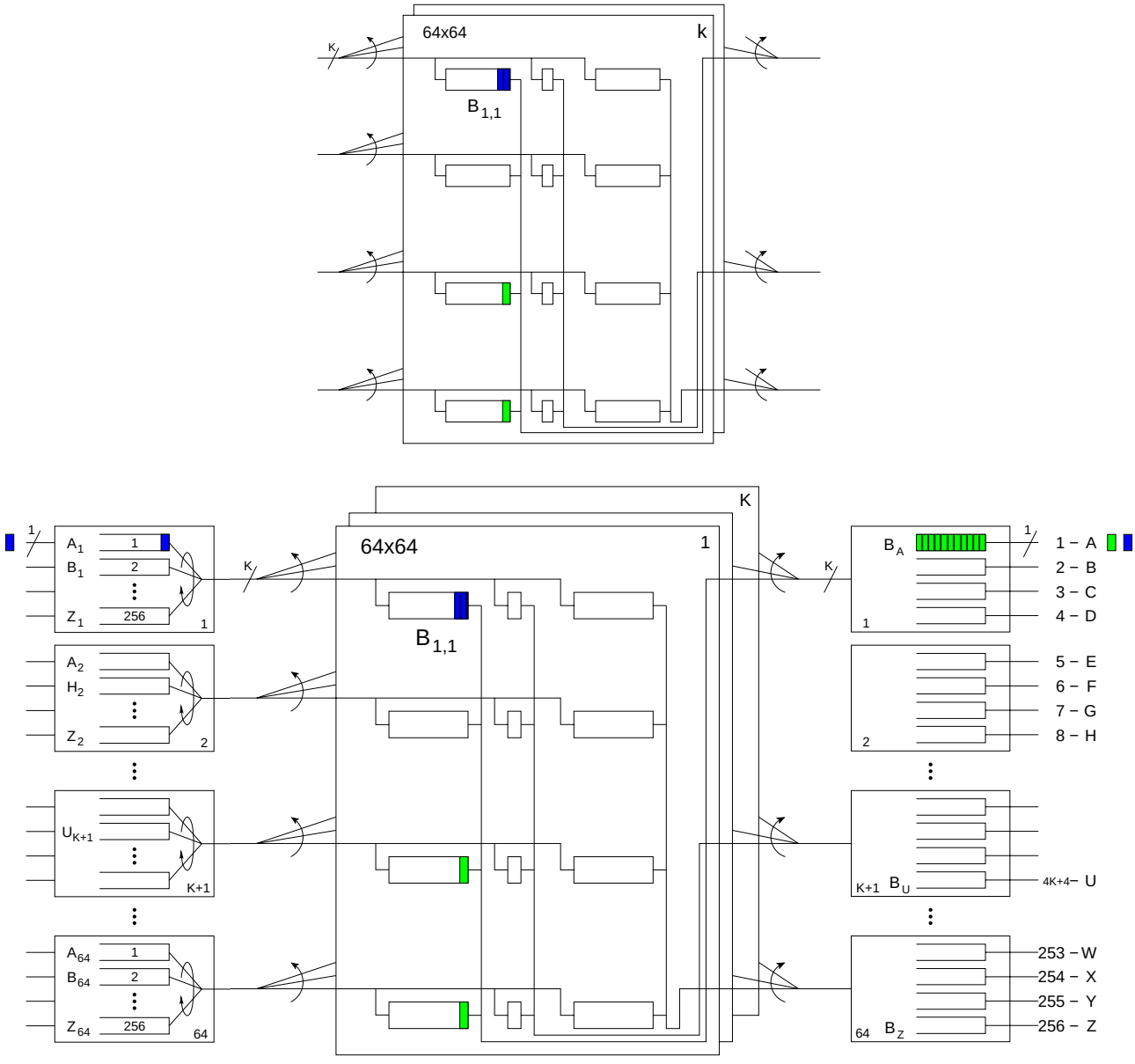

Fig. 30. Step 3: Blue packets accumulate in buffers $B_{1,1}$.

► **Step 4:**

If $B_r < N$, buffers $B_{1,1}$ fill up before all the green packets are transferred from the planes to output buffer B_A. The arrival of blue packets stops at the first input subport until all the green packets are transferred to output buffer B_A. The arrival pattern now changes as follows. The first packet of a new flow of packets (colored red) destined to subport 1-A arrives at the first input subport of the first input adaptor. The packets of this flow are assumed to arrive in at a constant rate of one packet every second time slot. At the first subport of each of adaptors 2 through 64 the first packet of new flow of packets (colored brown) destined to subport 1-A arrive as shown in Figure 31.

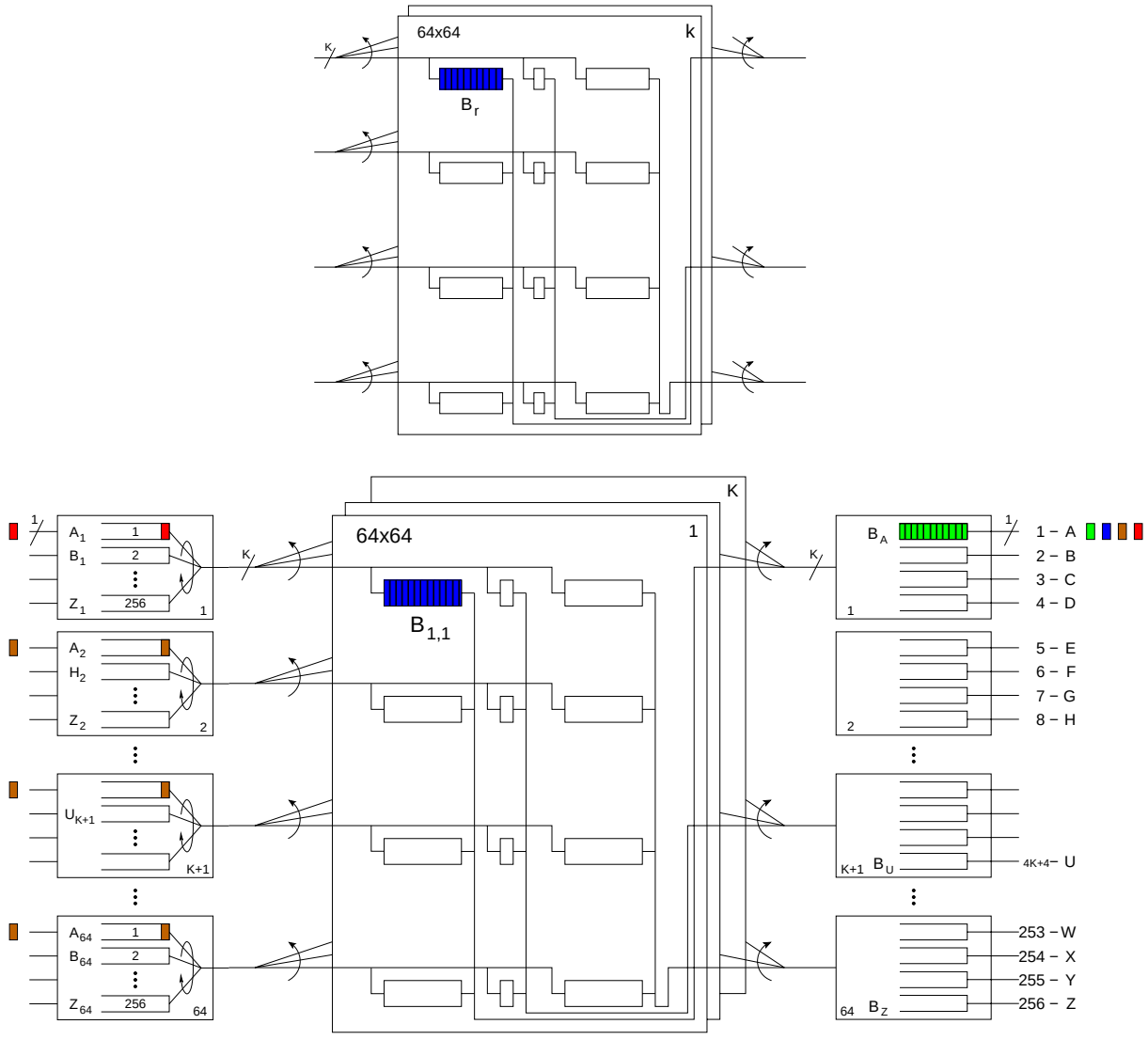

Fig. 31. Step 4: Arrival pattern changes.

► **Step 5:**

One packet place becomes available in $B_{1,1}$ of the first plane because of the transfer of the first blue packet to output buffer B_A. The red packet is transferred to buffer $B_{1,1}$ of the first plane. Also the brown packets are transferred from the input adaptors to buffers $B_{1,1}$ of the first plane as shown in Figure 32.

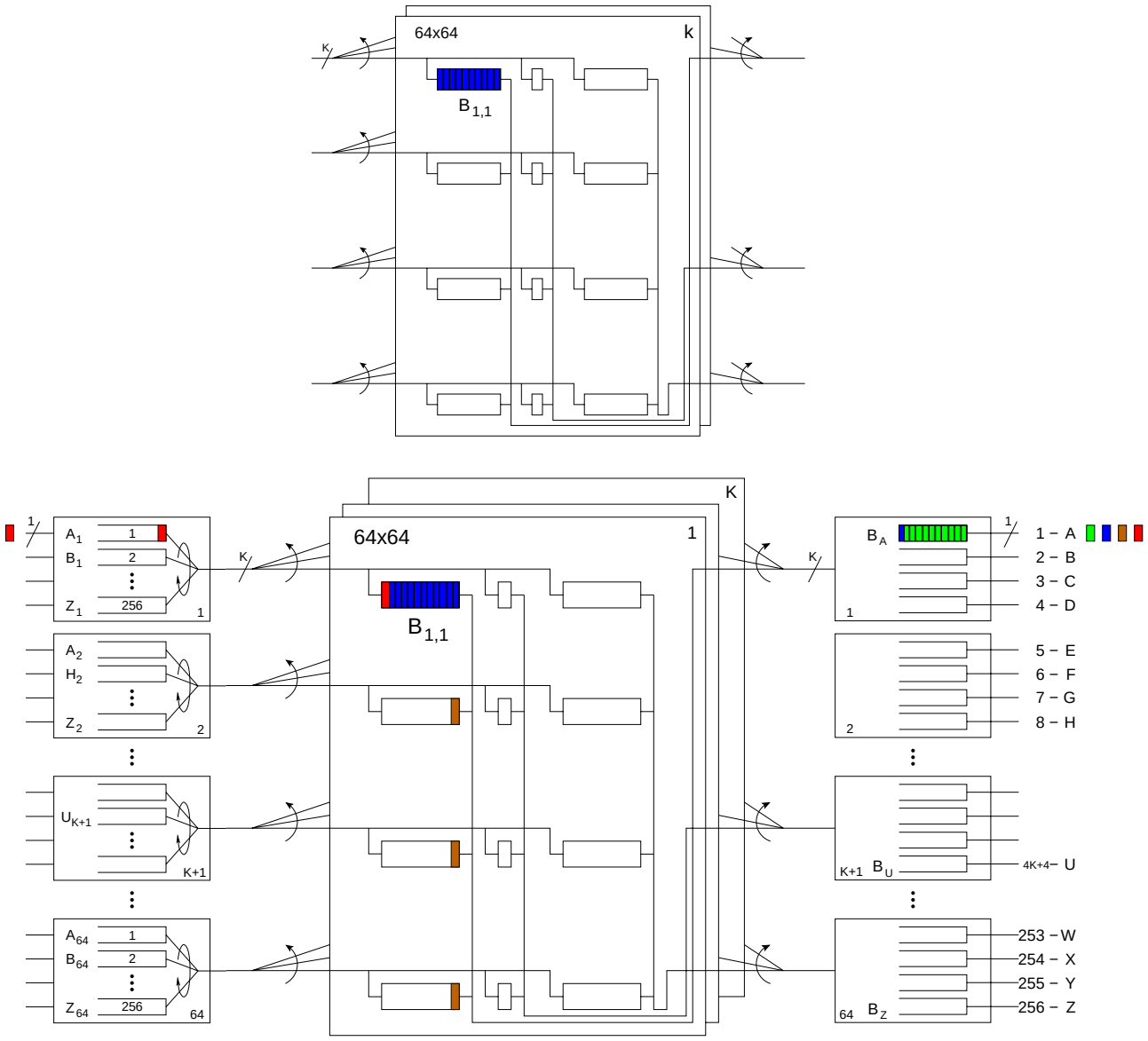

Fig. 32. Step 5: First red packet and brown packets are transferred to plane 1.

‘

► **Step 6:**

Brown packets are successively transferred from buffers $B_{2,1}, B_{3,1}, \ldots$ of the first plane to output buffer B_A. For every brown packet transferred from the first plane, the round-robin scheme results in $K-1$ subsequent blue packet transfers from buffers $B_{1,1}$ of planes 2 through K. At the same time red packets are accumulated in buffers $B_{1,1}$ of the various planes as shown in Figure 33. At the time when buffers $B_{1,1}$ of planes 2 through K are depleted from blue packets, buffers $B_{2,1}, \ldots, B_{B_r,1}$ of the first plane are emptied from brown packets. At that instant there are $B_r - 1$ blue packets in buffer $B_{1,1}$ of the first plane and $N - B_r$ brown packets in buffers $B_{B_r+1,1}, \ldots, B_{64,1}$ of the first plane. Therefore, the total number of blue and brown packets in the first plane is equal to $N - 1$.

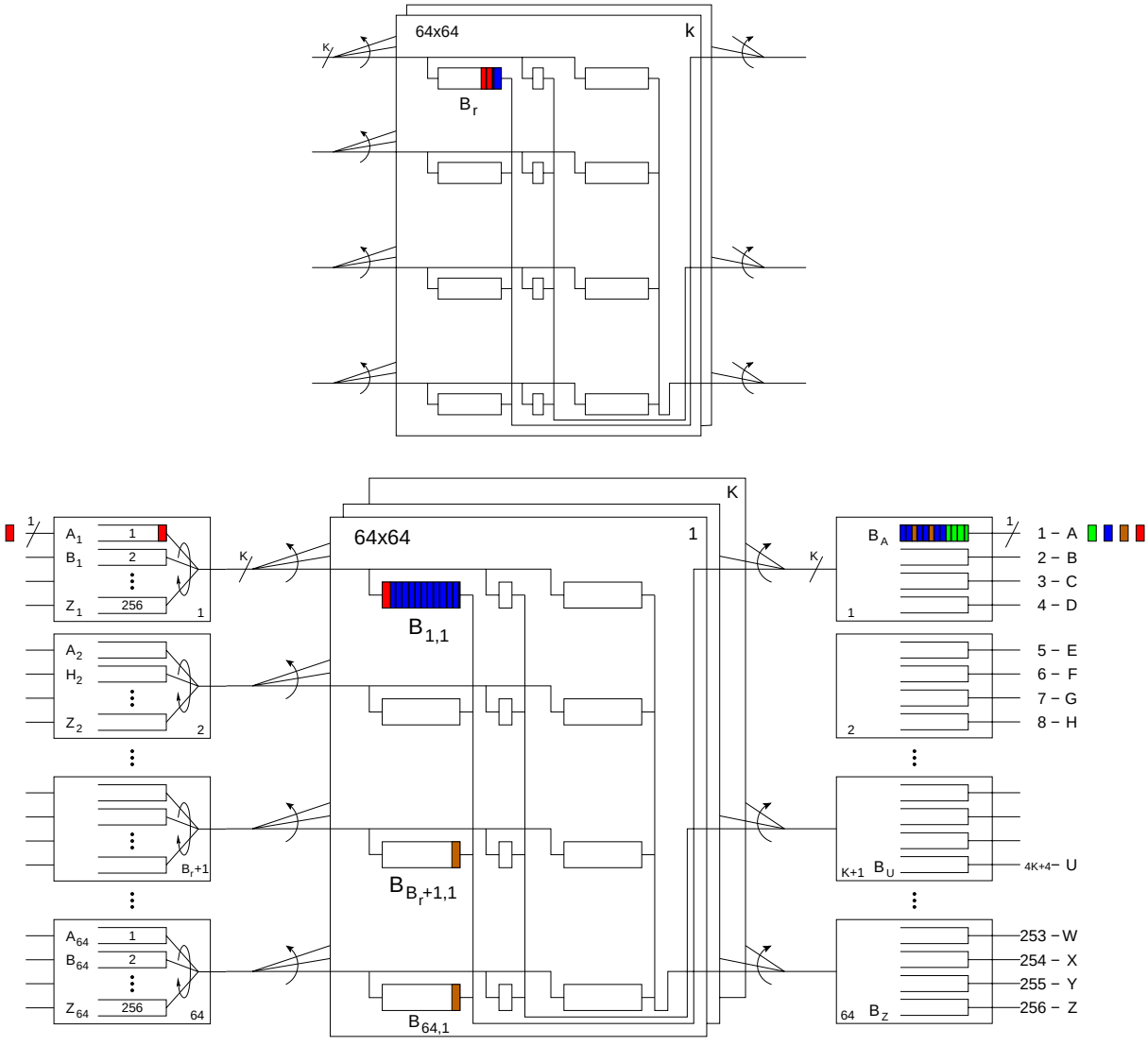

Fig. 33. Step 6: Buffers $B_{1,1}$ of planes 2 through K are depleted.

► **Step 7:**

Buffer B_A is always full as it is emptied at a rate of one packet per time slot and is fed by one packet per time slot from the planes. For every brown or blue packet transferred from the first plane, and as there is a sufficient number of red packets stored in planes 2 through K, the round-robin scheme results in $K-1$ subsequent red packet transfers from planes 2 through K. These red packets are stored in the resequence queue as they all have to wait for the first arrived red packet stored in buffer $B_{1,1}$ of the first plane. The first arrived red packet stored in buffer $B_{1,1}$ of the first plane is transferred to buffer B_A only after all the $N-1$ brown and blue packets have been transferred to buffer B_A, as shown in Figure 34. At that instant the number of red packets stored in the resequence queue is equal to $(K-1)(N-1)$.

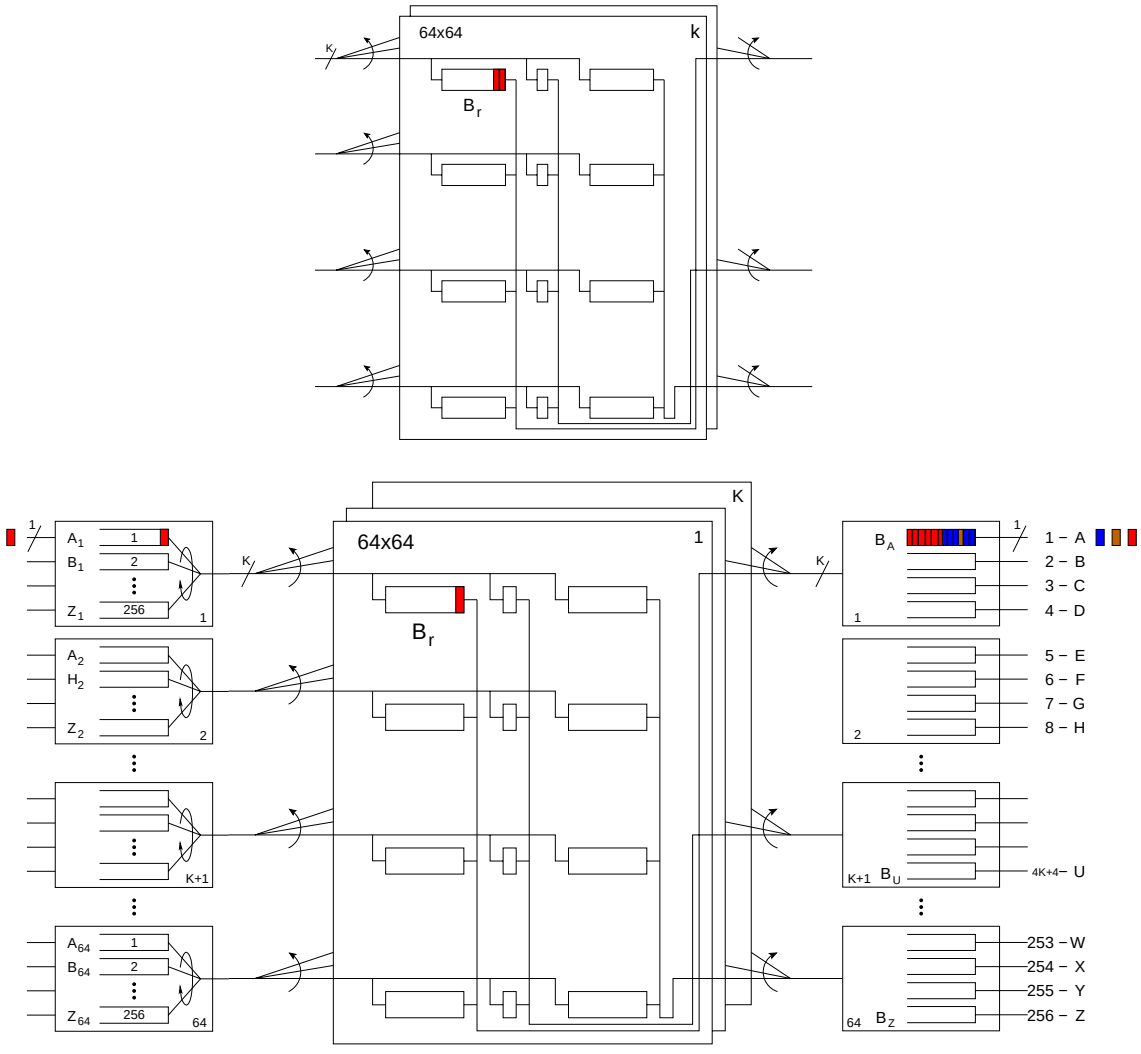

Fig. 34. Step 7: Packets are continuously dispatched to buffer B_A.

Remark 10. The maximum packet rate observed at any of the switch links is equal to one packet every K time slots. Moreover, for the queues to build up in Steps 1, it should hold that $Ks > 1$, or $s > 1/K$. Consequently, the scenario presented holds for any value of the switch port speed s in the range $s > s_{\min} = 1/K$.

Remark 11. The scenario presented obtains the worst case of resequence queue size by considering traffic present on only one subport of any given input or output adaptor. Furthermore, this size cannot be increased by considering more than one subports. Consequently, the number of subports of each adaptor is irrelevant, as it does not affect the outcome.

Remark 12. It turns out that the results obtained hold also in the case where $B_r > N$.

APPENDIX D
WORST-CASE RESEQUENCING FOR LOW-PRIORITY TRAFFIC

Here we present a sequence of events that lead to the worst-case resequencing for low-priority traffic.

► Step 1:

We start by considering hot spot traffic arriving at the first subport of each of the first $K + 1$ input adaptors and destined to output subport A. The frame arriving at the first adaptor (colored red in Figure 35) is assumed to be of a low priority, whereas the remaining frames arriving at adaptors 2 through $K + 1$ (colored green in Figure 35) are assumed to be of a higher priority. It is assumed that the low-priority packets arrive in a constant stream of one packet per time slot, whereas the packets of the high-priority frames arrive at a rate of one packet every K time slots.

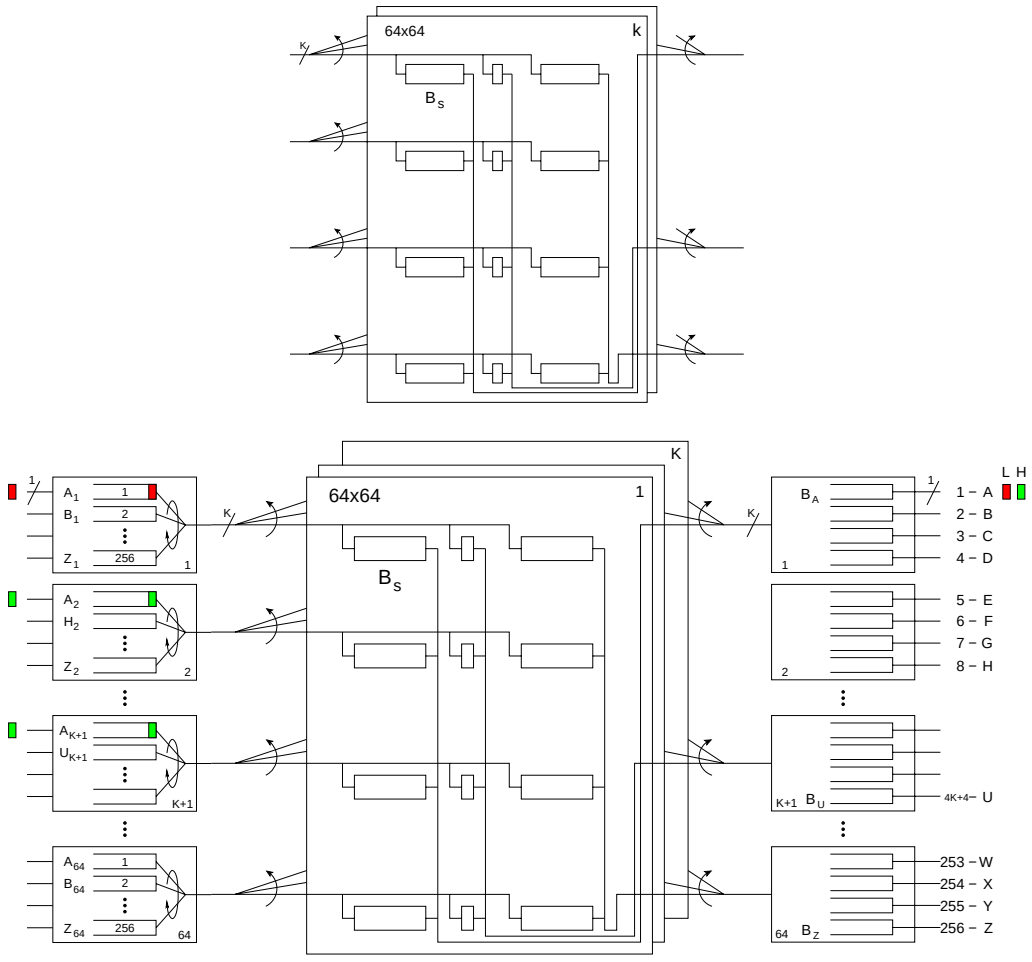

Fig. 35. Step 1: Subport A is a hot spot.

► **Step 2:**

It is further assumed that all $K+1$ packets are subsequently transferred to the first plane and more specifically to buffers $B_{1,1},\ldots,B_{K+1,1}$ of the first plane as shown in Figure 36. As the K green packets have a higher priority than the red packet, they will be transferred first out of the first plane. At the same time, the arrival pattern changes as follows. At the first subport of each of adaptors 2 through $K+1$ new frames arrive destined to the fourth subport of the corresponding output adaptors 2 through $K+1$, respectively. Thus, at adaptors 2 and $K+1$ the arriving packets are placed at the VOQs H_2 and U_{K+1}, respectively. These packets (colored brown in Figure 36) are assumed to arrive at a rate of one packet per time slot, except when interrupted by packets of the green frames destined to subport A.

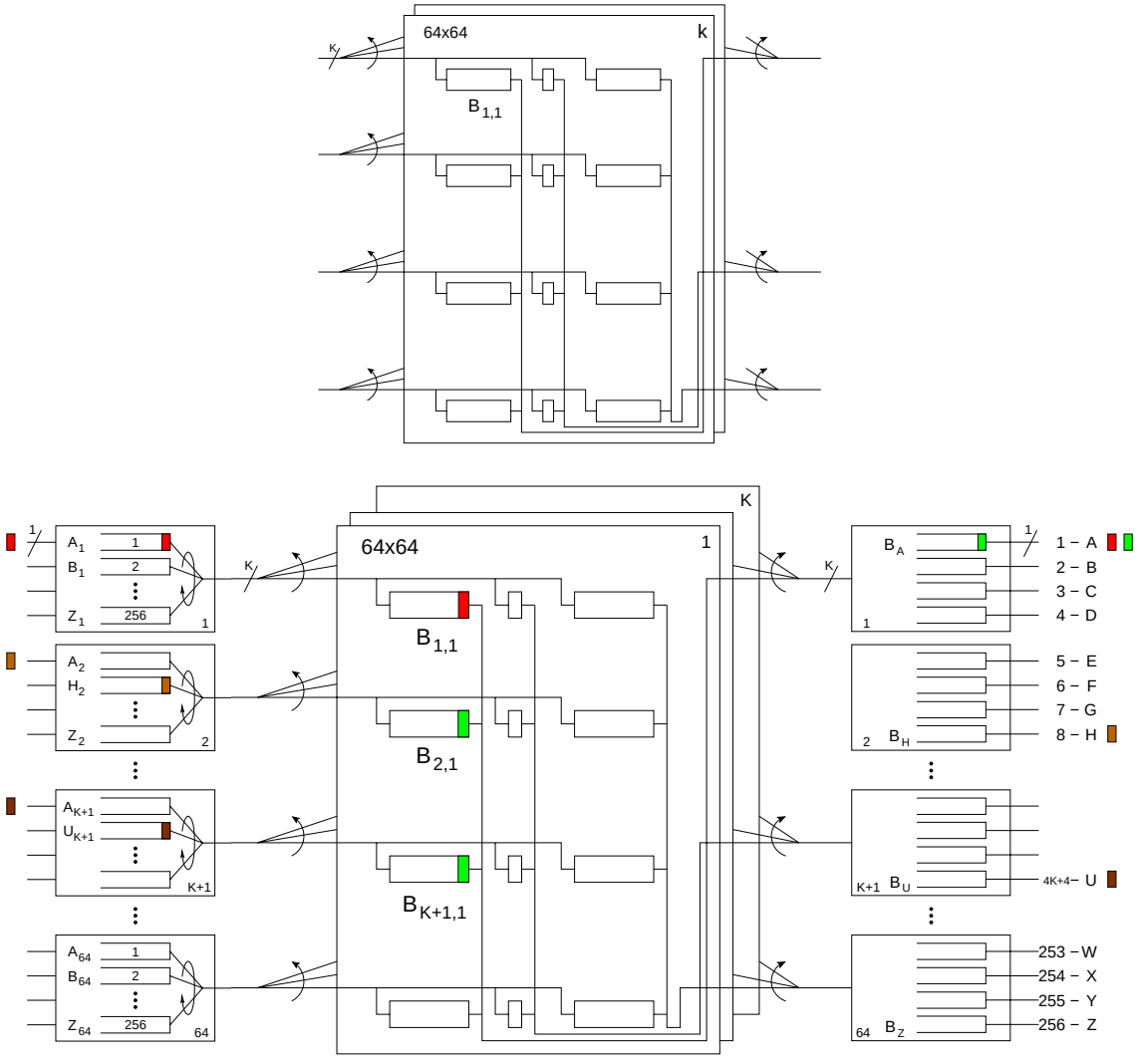

Fig. 36. Step 2: First packets of new frames arrive.

► **Step 3:**

In the next $K-1$ time slots, the red and brown packets are successively transferred to planes 2 through K and then to the corresponding output adaptors as shown in Figure 37. The red packets are stored in the resequence queue of buffer B_A because they all have to wait for the red packet located in buffer $B_{1,1}$ of the first plane. Also, during this period of $K-1$ time slots, $K-1$ green packets are successively transferred from buffers $B_{2,1}, \ldots, B_{K+1,1}$ of the first plane to buffer B_A and subsequently transmitted at the output subport A.

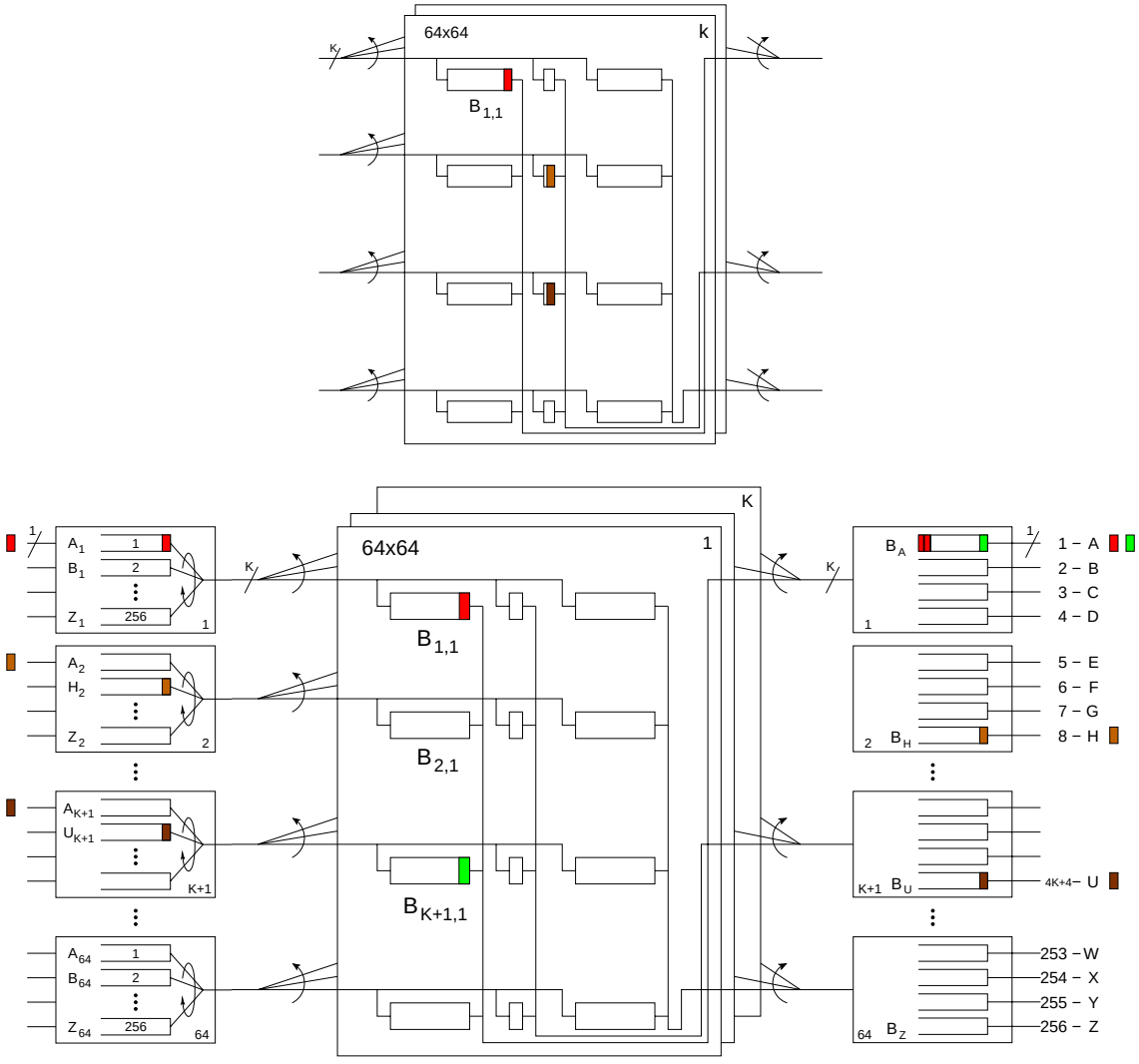

Fig. 37. Step 3: Packets are transferred to the planes.

► **Step 4:**

The cycle of K time slots is repeated with green packets arriving at the input adaptors 2 through $K + 1$. Furthermore, the stream of red packets is interrupted by the first packet (colored blue in Figure 38) of a frame arriving at the first input subport of the first input adaptor and destined to subport 256-Z. This packet is therefore placed at VOQ Z_1. The blue packets are assumed to arrive at a rate of one packet every K time slots in a way that they are all transferred to the first plane, whereas all the packets (except the first one) of the red flow are transferred to the remaining planes.

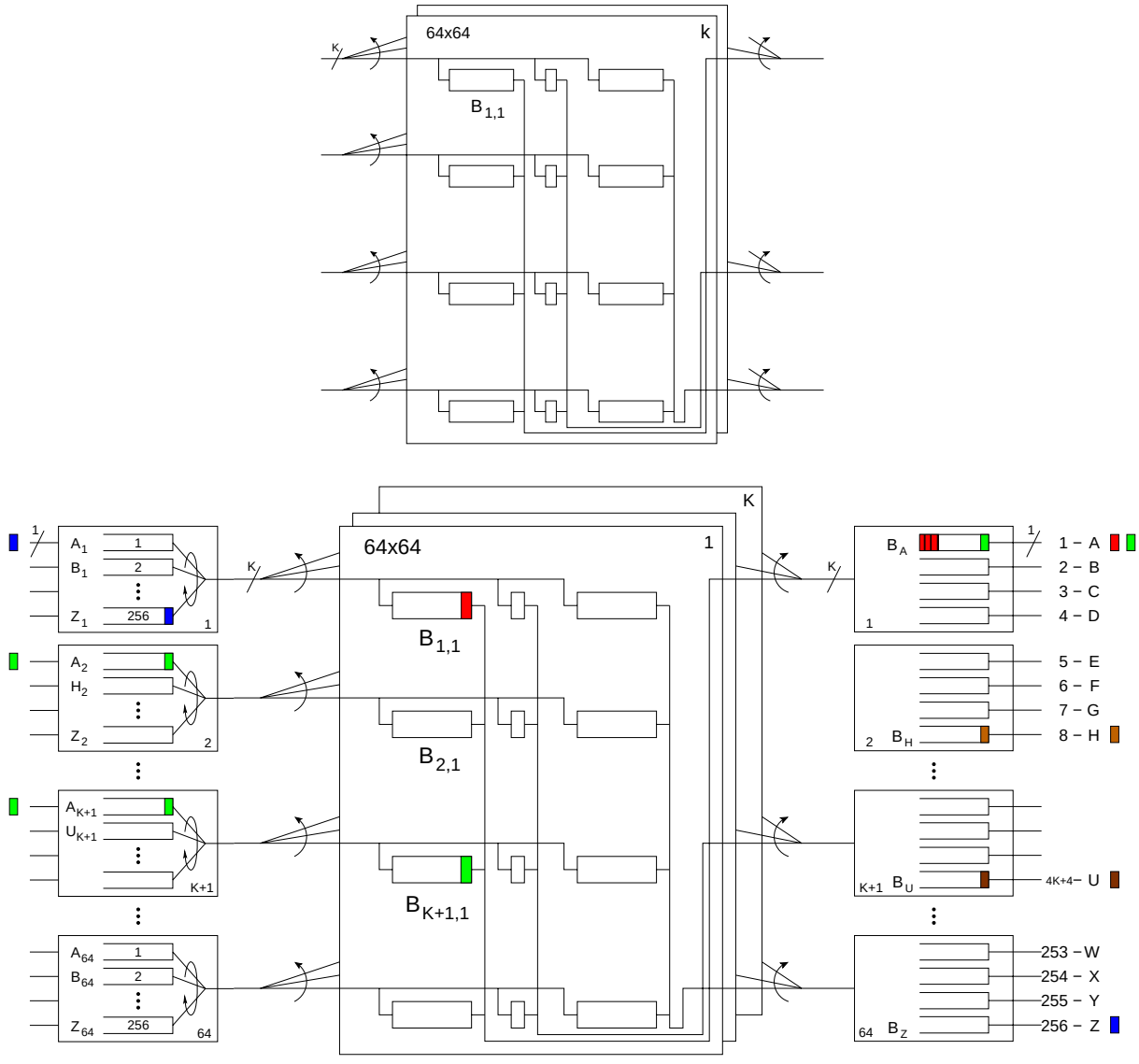

Fig. 38. Step 4: First packet of frame arrives in the first subport.

► **Step 5:**

According to the periodicity of the plane load balancing schemes, all the packets of the input adaptors will be transferred in the next time slot to plane 1. Consequently, as the last green packet in buffer $B_{K+1,1}$ of the first plane is transferred to buffer B_A, a new batch of K green packets are transferred from the input adaptors to buffers $B_{2,1}, \ldots, B_{K+1,1}$ of the first plane as shown in Figure 39. Also the blue packet is transferred from VOQ Z_1 to buffer $B_{1,64}$ of the first plane. The pattern of packet arrivals during time slots 2 through K of the previous cycle is repeated.

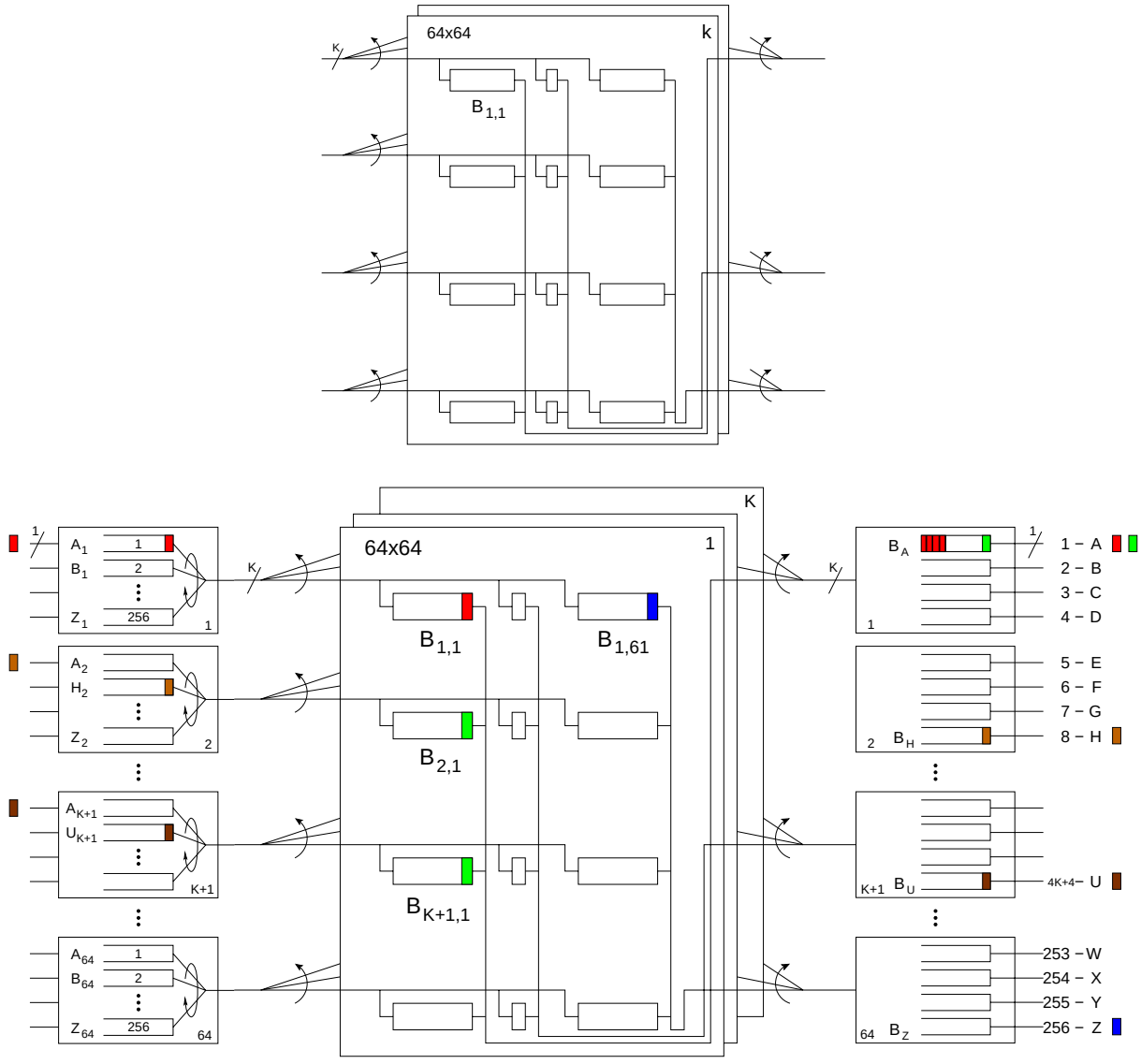

Fig. 39. Step 5: Packets are transferred to plane 1.

► **Step 6:**

In the next time slot, the red and brown packets are transferred to plane 2; the blue packet is transferred from plane 1 to buffer B_Z of the last output adaptor, and a green packet is transferred from buffer $B_{2,1}$ of the first plane to buffer B_A. There are no red or brown packets transferred to the output adaptors.

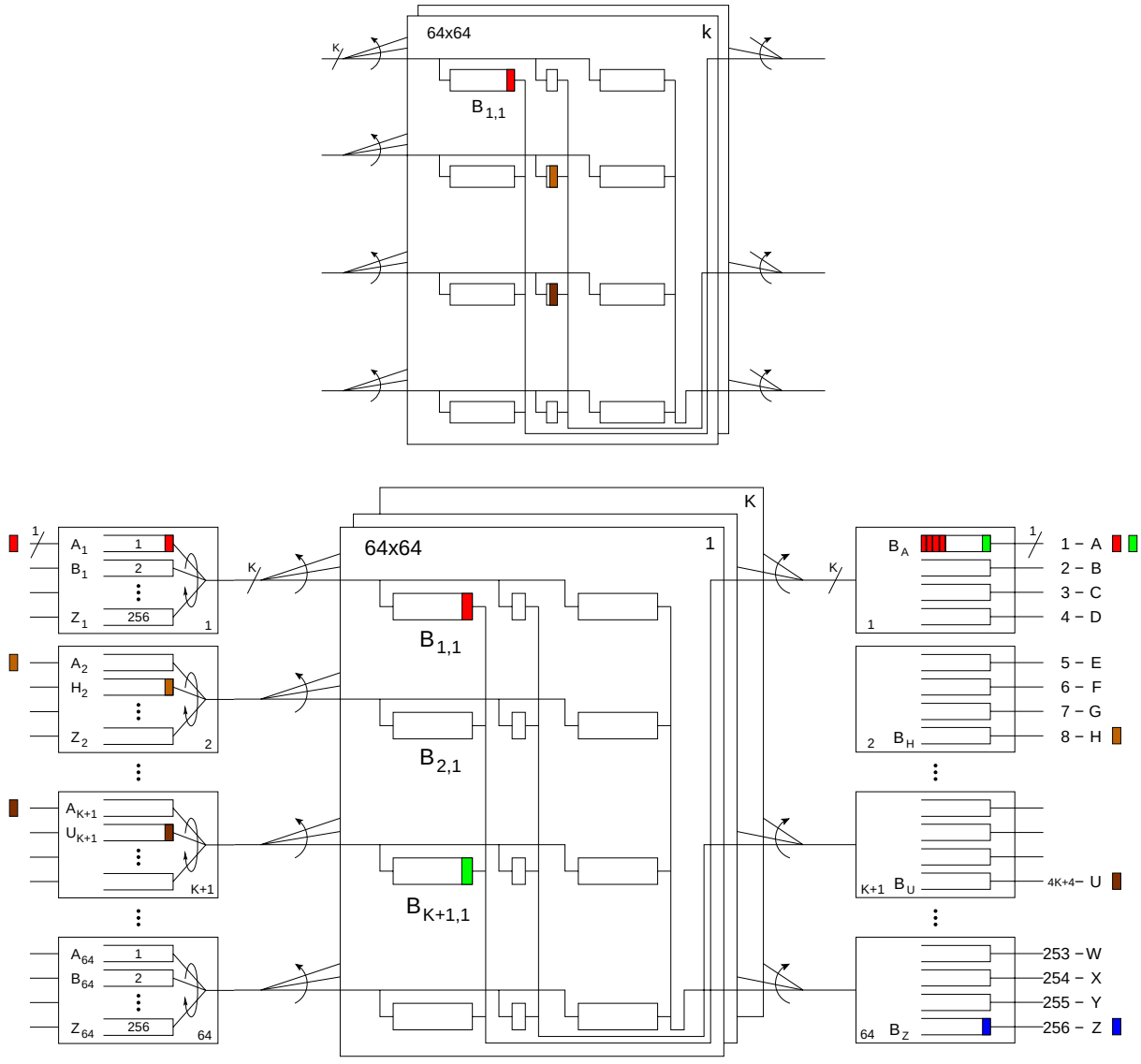

Fig. 40. Step 6: Packets are transferred to plane 2.

► **Step 7:**

In the next $K-2$ time slots, the red and brown packets are successively transferred to planes 3 through K and then to the corresponding output adaptors as shown in Figure 41. The red packets are stored in the resequence queue of buffer B_A because they all have to wait for the red packet located in buffer $B_{1,1}$ of the first plane. Also, during this period of $K-2$ time slots, $K-2$ green packets are transferred from $B_{3,1},\ldots,B_{K+1,1}$ of the first plane to buffer B_A and subsequently transmitted at the output subport A.

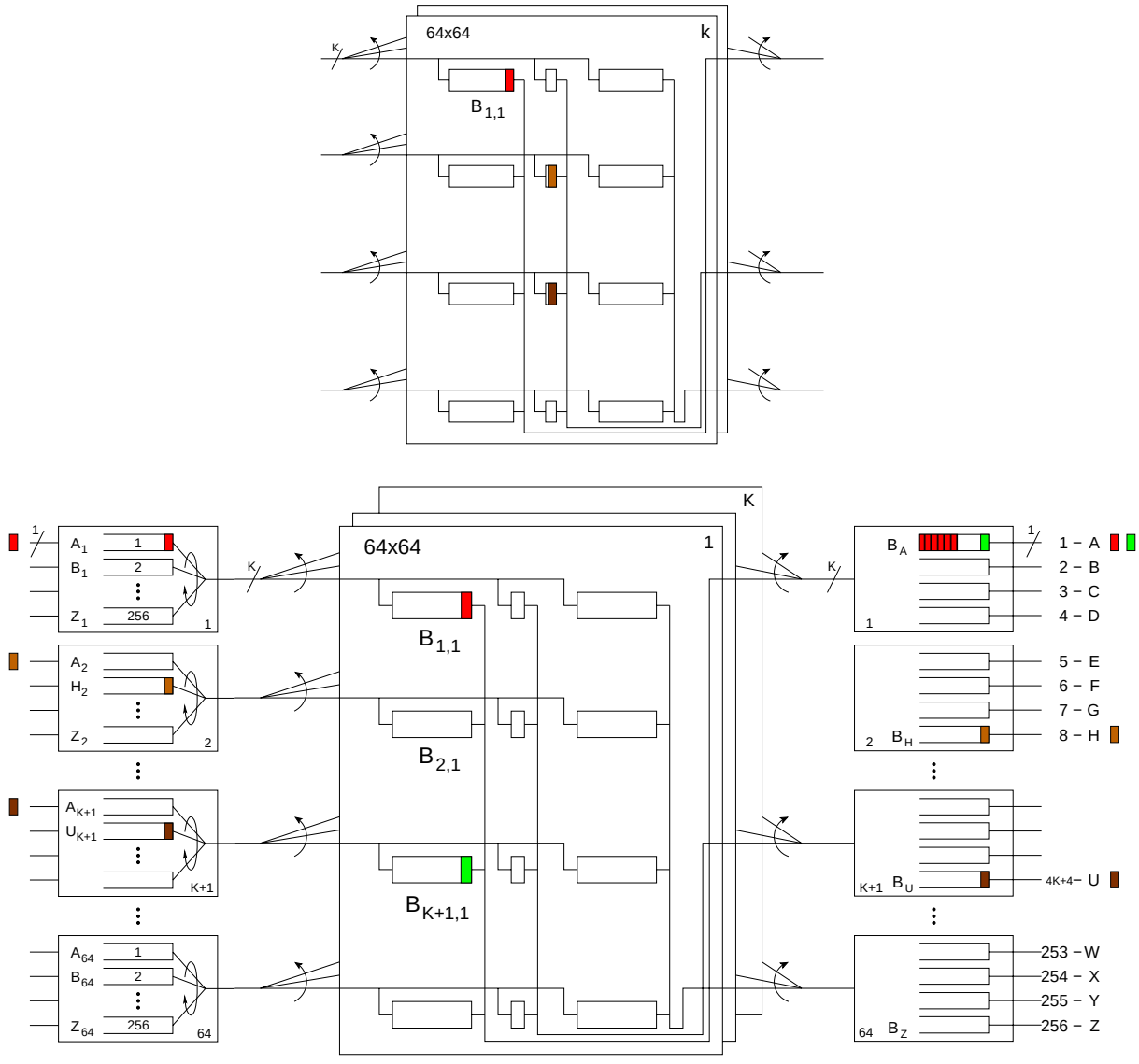

Fig. 41. Step 7: The resequence queue in B_A grows constantly.

Clearly, repeatedly applying the K time slot cycle described in Steps 4 through 7 results in the constant increase of the number of red packets stored in the resequence queue of buffer B_A because the red packet in buffer $B_{1,1}$ of the first plane remains blocked and it is therefore never transferred to buffer B_A.

Remark 13. The maximum packet rate observed at any of the switch links for steps 1 through 7 is equal to one packet every K time slots. Consequently, the scenario presented holds for any value of the switch port speed s in the range $s > s_{\min} = 1/K$.

Remark 14. The scenario presented obtains the worst case of resequence queue size by considering traffic present on only one subport of any given input or output adaptor. Furthermore, this size cannot be increased by considering more than one subports. Consequently, the number of subports of each adaptor is irrelevant, as it does not affect the outcome.

Remark 15. The scenario presented can be modified in a way such that the stream of red packets is directed only to a single plane. This is achieved by appropriately increasing the rate of blue packets to $K - 1$ packets every K slots and decreasing the rate of red packets to one red packet every K slots.

Remark 16. The buffer occupancies in the scenario presented are at most one packet. Consequently, both the round-robin and state-dependent plane load balancing schemes behave identically.

REFERENCES

[1] S. Iyer, A. Awadallah, and N. McKeown, “Analysis of a packet switch with memories running slower than the line rate”, in *Proc. IEEE INFOCOM ’00*, vol. 2, pp. 529-537, Tel Aviv, Israel, March 2000.

[2] F. Abel, C. Minkenberg, R. P. Luijten, M. Gusat, and I. Iliadis, “A four-terabit single-stage packet switch with large round-trip time support,” to appear in *Proc. Hot Interconnects 10*, Stanford University, August 2002.

[3] I. Iliadis and Y.C. Lien, “Resequencing delay for a queueing system with two heterogeneous servers under a threshold-type scheduling”, *IEEE Trans. Communications*, vol. COM-36, no. 6, pp. 692-702, June 1988.