# Article information:

Implementation and testing of openflow switch using FPGA | IEEE Conference Publication | IEEE Xplore
<https://ieeexplore.ieee.org/document/8203995>

# Article summary:

1. Traditional network infrastructure consists of legacy switches and routers that are limited in their ability to handle increasing network traffic demands.

2. SDN (Software-Defined Networking) redefines the architecture by decoupling the intelligence (control plane) of the switch from the forwarding mechanism (data planes), enabling remote network administrators to easily change and reinstall policies to meet packet traffic demands.

3. An OpenFlow switch was implemented on FPGA using a modular approach, providing flexibility, scalability, and portability at line-rate performance. Future work includes implementing the design on FPGA and performing various OpenFlow compliance tests.

# Article rating:

Appears moderately imbalanced: The article provides some useful information, but is missing several important points or pieces of evidence that would be required to present the discussed topics in a balanced and reliable way. You are encouraged to seek a more balanced perspective on the presented issues by exploring the provided research topics and looking at different information sources.

# Article analysis:

The article discusses the implementation and testing of an OpenFlow switch using FPGA. It provides a detailed explanation of the traditional network infrastructure architecture and its limitations, as well as the benefits of SDN and OpenFlow in addressing these limitations. The article also describes the design and implementation of the OpenFlow switch, including its modules and their functions.

Overall, the article appears to be informative and well-researched. However, there are some potential biases and missing points of consideration that should be noted. For example, the article focuses primarily on the benefits of SDN and OpenFlow without discussing any potential risks or drawbacks. Additionally, while the article mentions that commercial solutions are inflexible and expensive, it does not provide any evidence or examples to support this claim.

Furthermore, the article may be biased towards promoting FPGA-based OpenFlow switches over other solutions. While it briefly mentions that dedicated OpenFlow switches can also be implemented by modifying existing legacy switches, it does not provide any further information or analysis on this approach. Additionally, while the article notes that there are limited studies on performance analysis of OpenFlow switches, it only discusses one related work without exploring any potential counterarguments or alternative perspectives.

In terms of missing evidence for claims made, the article states that "OpenFlow switch implemented on Linux based PC outperforms others" but does not provide any data or sources to support this claim. Similarly, while the article mentions future work on managing TCAMs efficiently and segregating traffic related to MPLS/IPV6/ARP, it does not explain why these tasks are important or how they will improve the functionality of the OpenFlow switch.

Overall, while the article provides a useful overview of implementing an OpenFlow switch using FPGA, readers should be aware of potential biases and missing points of consideration when evaluating its content.

# Topics for further research:

* Potential risks and drawbacks of SDN and OpenFlow
* Comparison of FPGA-based OpenFlow switches with other solutions
* Advantages and disadvantages of modifying existing legacy switches for OpenFlow
* Performance analysis of OpenFlow switches: alternative perspectives and counterarguments
* Data and sources supporting the claim that OpenFlow switch on Linux-based PC outperforms others
* Importance and benefits of managing TCAMs efficiently and segregating traffic related to MPLS/IPV6/ARP in OpenFlow switches.

# Report location:

<https://www.fullpicture.app/item/53c4cbcf89c753e016970d17ae8fb940>