*International Journal of Engineering & Technology*, 7 (4.40) (2018) 94-97

**International Journal of Engineering & Technology**

*Website: www.sciencepubco.com/index.php/IJET*

***Research paper***

# Implementation of FPGA in Index Data Storage as A Database

**Ferry Wahyu Wibowo[1]***

*[1]Informatics Department, Universitas Amikom Yogyakarta*
**Corresponding author E-mail: ferry.w@amikom.ac.id*

**Abstract**

Nowadays, the database applications have been built on the top of file systems. Although this system is very tough it has a side that should be aware and concerned. If the database doesn't manage well it will cause some problems i.e. data redundancy and inconsistency, difficult to access data, data isolation, and integrity problems. The data inconsistency could be emerged by duplication of information from one or many different files or multiple file formats. the hardware-based implementation provides benefits in modifying and reconfiguring, meanwhile, the acceleration using this approaching has a good performance than software-based. In major cases, both of them using a bridge to connect each other which is called co-hardware/software.

***Keywords***: *Database, FPGA, Index, Storage.*

## 1. Introduction

The electronic data has to be made because some reasons which are ease of use, sharing, analyzing, and the ability to store a large volume in a small space. The term of the database is a collection of the files that are used to store data. In the application of the software it can be easy to understand but in the aspect of the hardware, the function is the same with how the database in software does. The applications of the database have touched and spread all aspects of information. Meantime, there is another term which is called a database management system (DBMS). A DBMS is a C program that allows the user to manage a large database efficiently and stored at a long time. There are examples of the DBMS e.g. Oracle, DB2, Sybase, SQL Server, etc. The main functions of the database consist of create/store, search, update, etc. The DBMS consists of information and provides an environment in both efficient and convenient so the usage of this system is very effective in managing data.

Nowadays, the database applications have been built on the top of file systems [1]. Although this system is very tough it has a side that should be aware and concerned. If the database doesn't manage well it will cause some problems i.e. data redundancy and inconsistency, difficult to access data, data isolation, and integrity problems. The data inconsistency could be emerged by duplication of information from one or many different files or multiple file formats. The usage of the database also needs to be defined in writing a new program to obtain different tasks while when it goes to the integrity of huge data in the database, it should become part of the program code and it is hard to change an existing data in the database. But with some technique, it could be easy to take control of invoking the data through the database [2]. The wrong case in input and making a database, it may leave a database in inconsistent conditional and make the database can't be updated.

The database is a fundamental aspect of presenting the information. This information is very needed to the user in performing the model of the interfaces. The design of the database is used for the designer to make a concept, relational, and normalization to ease modeling. There are 3 abstraction levels due to the database, namely physical level, logical/conceptual level, and view level. The physical level is figuring level of how a record is stored, whereas the logical/conceptual level is describing data stored in the database and the relationships among the data while a view level is a program of an application that hides the data types and also information for the security reasons. In performing the database, it is similar to types and variables in programming languages but it has an ability to modify the physical schema without changing the logical schema. The interfaces between the levels and components have to be described at the beginning so it won't be a problem when the data have been changed in some parts. The file management systems are absolutely important to provide a well-performing data in extracting and sharing of files. That's way, the relationships of each data must be implemented in the application code.

There are some of the distribution strategies for databases which consist of centralized data and processing, client-server, distributed database, distributed logic, and intelligent terminal [3]. The centralized data and processing use terminal with some codes to do work on it. The client-server is usually maintained on a server and processed on the client workstation. The distributed database is sharing and distributing data among different locations and processing data is done in where the data exist so in this case generally it will be partitioned. The distributed logic is using a data storage that is distributed and processed at the optimal location. The intelligent terminal is using data and processing centralized but it uses remote devices to control. The structure of the database could be optimized to gain the optimal storage [4].

In the concept of the computer, there are two main parts those influence in the work principles i.e. hardware and software. These aspects although have different implementations but it supports each other [5]. When talking about the hardware-based, it is usually imagining with the physical aspect that should exist in the real world. It can't be blamed the way of this definition but in this case, it could be defined as a process which the hardware-based has parallel processing. However, the hardware-based implementation provides benefits in modifying and reconfiguring, meanwhile, the acceleration using this approaching has a good performance than software-based. In major cases, both of them using a

bridge to connect each other which is called co-hardware/software. The issue of making software-based implementation into hardware-based implementation has a class as an interesting method. The strategy in implementation is not the same between them so it depends what kind of the implementation that will be resulted. One of the devices in the hardware platform is field programmable gate arrays (FPGAs). In reconfiguring the FPGAs, the hardware designers have three design goals i.e. runtime performance, flexibility, and reconfiguration speed [6]. Each of them sometimes can't meet the requirements but it can be considered to reach that approaching. With the ability of the FPGAs, this paper aims at the prototyping of the index data storage like a database. In this case, it will analyze how the database behavioral can be built using the hardware-based and how it goes on the aspect of the design.

## 2. Co-Hardware/Software-Based Database

A database is a group of the tables or numbers of related files. Many models of the database that could be made, one of the database model is a relational database. This model is based on the n-ary relational concept. In the relational database, one table is related to the one relation. Each column in the table is called attribute, while each row of the data in the table expresses a record and each attribute expresses a field. A field is a group of characters that form one or group of words or identity number. The field consists of bytes those are a group of bit numbers representing a character. The domain of the attribute is a set of all of the attribute members existed. Physically, the database is a group of files, while the file is a group of records, and each record consists of numbers of fields. A special attribute on the table identifies a relation element uniquely is known as a key. The hierarchy of the database that is employed in the software-based can be seen in figure 1.

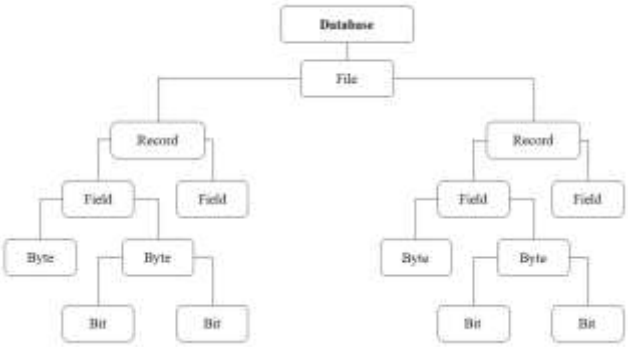



**Fig. 1:** Hierarchy of data

The database management system (DBMS) is employed to make an organized data so it will be useful when it is processed and performed. The application built is capable to define and manipulate data. There are 2 main applications namely data definition language (DDL) and data manipulation language (DML). The DDL is applied to define the database, whereas the DML is applied to do inserting, updating, retrieving, deleting, etc. The DDL is instructions used by the administrator of the database to defining scheme and sub-scheme of the database. The result of the DDL compilation is stored in the system catalog that combines a metadata. The DML contains 2 types i.e. Procedural DML and Non-procedural DML. The Procedural DML is instructions to determine what kind of data and how to obtain the data, while the non-procedural DML is instructions to determine what data needed without mention how to obtain it. To get this is by typing the instructions that are used to call the DBMS in manipulating a record. The system database components consist of data, users, procedures, hardware, and software. The data are stored in the storage media which are separate between the entity and the program that is used to access it. The user, in this case, has 2 meanings namely end user and application program those are access and process data. Procedures are rules which are defined and followed by the users. Hardware is the physical computer system while the software is the program that works to access and process the data. The architecture of the database contains internal level, conceptual level, and external level. This architecture is shown in figure 2.

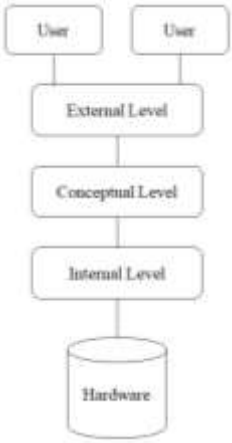
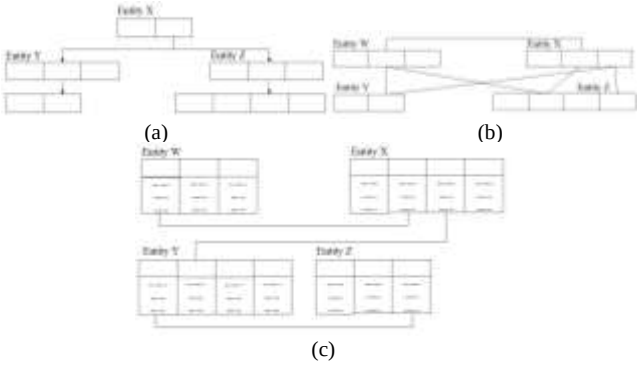



**Fig. 2:** Architecture of the database

In this architecture, the internal level interacts with the hardware whilst the external level interacts with the users. The conceptual is level is the mediator between the external level and the internal level which define the logical view of the data. The concept of the database is a logical integration of the records from storage where one another is correlated. This concept is to eliminate the data redundancy in implementing the storage space inefficiently and making slow the record management, inconsistency data, and duplicated data. So it will gain the data independence when it is done changing in the data structures so the program won't be changed because the program that processes the data is separated with the data stored in the database.

Some operations used in the database consists of an insert, delete, update, select, project, join, union, intersection, and difference. The instruction of the insert is employed to insert a new tuple into the relation. A tuple is a group of an attribute value. The instruction of the delete is employed to eliminate tuple from the relation. The instruction of the update is employed to change the values of some attributes of a tuple. The instruction of the select employs some criteria to select some tuples from the relation. The instruction of the project is creating a relationship with fewer attributes. The instruction of the join is used to combine the relations based on a common attribute. The instruction of the union is combining the relation for each tuple. The instruction of the intersection is creating a new relation on each tuple in both relations. The instruction of the difference is creating a new relation in the first table but not in the second one. There are 3 database models i.e. hierarchical model, network model, and relational models (see figure 3).



(a) (b)

(c)

**Fig. 3:** (a) Hierarchical model, (b) Network model, (c) Relational model

The hierarchical model organizes data as a tree which each entity has a parent with many children. This model has been abandoned now and not so well used. The network model has also abandoned.

This model depicts the organized entities in a graph. A graph consists of vertexes and edges. The relational model organizes data within two-dimensional tables. These tables have a relation to each other. This model is a common model used today.

## 3. Database Model Using FPGAs

FPGAs are a hardware device that has a parallel platform related to the programmable logic, routing, memory, and input/output (I/O). The contexts of the FPGAs are utilizing in prototyping gate array, complete systems, and implementing a simulation of the hardware design. It is much working on the reconfigurable systems. In the FPGAs, logic is configured by values that are stored in the static random access memory (SRAM) cells. The configurable logic blocks (CLBs) implement logic in SRAM that contains the truth tables. Not only that, but the CLBs also employ the multiplexers to control the SRAM. The CLB's routing applies transistor to make a connection or breaking among wires. The design flow of FPGA contains 6 steps i.e. specification, register transfer logic (RTL) design, functional simulation, synthesis & implementation, timing simulation, and device programming.

The implementation of the FPGAs has challenges in the future technology because of the FPGA's abilities [7]. However, the technology is always developed along with the time including the technology of FPGA. The beneficial due to the implementation of the FPGAs can be reconfigured at runtime with the number of systems in obtaining the intelligent task so it will be efficient in utilizing [8]. Some prototypes had been implemented in a PCIEe-attached FPGA system and integrated with the DBMS platform [9]. The paradigms of both RDBMS and FPGA could bridge the power in establishing the database analytics procedures so the complexity of the co-hardware/software design could be achieved [10].

## 4. Discussion

This paper conducts to implement the database based on the FPGAs. The design referred to the relational database which implements tables. For data input/output could be configured in some values. The principle of this design is storing data in the design through data input and the data can be updated via its address in the table. To push the data out as the record in the database, it could be chosen the data address. The attribute in this database table consists of address and data. For every added attribute could implement one design. This design can be shown in figure 4.

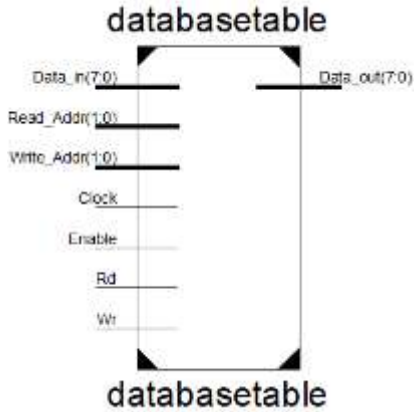



**Fig. 4:** Table design

In every input or output of the data, it should be enabled by port Enable in logic high. After it is enabled, the data input could activate pin of Wr and for the data output could activate pin of Rd. To assign the address could use a pin of Write_Addr while to choose data out address could use a pin of Read_Addr. The data input and output implemented in figure 4 is 8-bits data. In making some attributes in the tables can be parallelized. The register transfer logic (RTL) that is generated shown in figure 5.

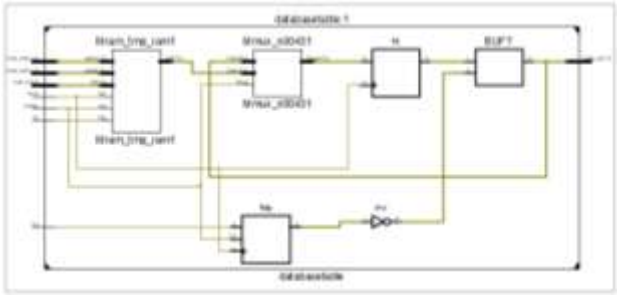


**Fig. 5:** RTL of table design

From figure 5, the RTL for this design consists of random access memory (RAM) as the main component. From the variation of this database table design ports which are configured as 8-bits, 16-bits, 32-bits, and 64-bits can be seen in table 1. The FPGAs device deployed is Xilinx Artix-7 FPGA (xc7a100t). The logic utilization consists of slice registers, slice look-up tables (slice LUTs), fully used look-up table flip-flop (LUT-FF) pairs, bonded input/output blocks (IOBs), and global clock buffer (BUFG).

**Table 1:** Utilization Summary

| No. | Component Consumptions | | | | |
|---|---|---|---|---|---|
| | Logic Names | Numbers (8-bits) | Number (16-bits) | Numbers (32-bits) | Numbers (64-bits) |
| 1 | Slice Registers | 9 (0%) | 17 (0%) | 33 (0%) | 65 (0%) |
| 2 | Slice LUTs | 18 (0%) | 34 (0%) | 58 (0%) | 114 (0%) |
| 3 | Fully used LUT-FF pairs | 8 (42%) | 16 (45%) | 32 (54%) | 64 (55%) |
| 4 | Bonded IOBs | 24 (11%) | 40 (19%) | 72 (34%) | 136 (64%) |
| 5 | BUFG/ BUFGCTRLs | 1 (3%) | 1 (3%) | 1 (3%) | 1 (3%) |

From the table 1 shows that the ports used are similar with the fully used LUT-FF pairs. The component consumptions of the slice registers and slice LUTs are not significant that the estimated values show 0% from the FPGA components have. To build a database can be connected in parallel (see figure 6). This table can be added as the design needs.

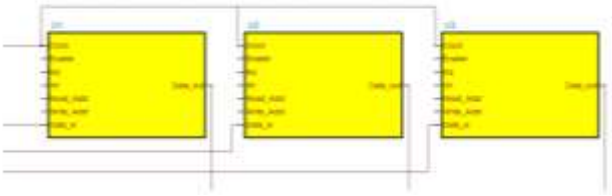


**Fig. 6:** Table connection

For making a join-like command as a database could use a concatenation block, whereas in making a select-like command as a database could use a multiplexer block. Many ways to design more complex design and for the hardware designer could have their own design.

## 5. Conclusion

This paper has shown the design of the database based on the FPGAs. This model has adopted the principle of the database based on the software. The design that has been proposed is fully hardware-based. The table that has been adopted is based on the relational model. The main component in this design is RAM that is depicted from the RTL design. The model could be generated and developed as the hardware designer wants. The comparison of consuming the components in FPGA has also been performed.

The future work for this design will be implemented on the clustering [11].

## Acknowledgment

This research has been funded by the ministry of technology research and higher education (RISTEKDIKTI) grant no. 019/HB-LIT/II/2018.

## References

[1] Manoj V, Comparative study of NoSQL document, column store databases and evaluation of Cassandra. International Journal of Database Management Systems (IJDMS), Vol. 6, No. 4, (2014), pp. 11-26, DOI: 10.5121/ijdms.2014.6402.

[2] Dahlan A & Wibowo FW, Transformation of data warehouse using snowflake scheme method. *International Journal of Simulation Systems, Science & Technology (IJSSST)*, Vol. 17, No. 35, (2016), pp. 16.1-16.10, available online: http://ijssst.info/Vol-17/No-35/paper16.pdf, last visit 03.08.2018.

[3] Madhuri DK, A novel approach for processing big data. International Journal of Database Management Systems (IJDMS), Vol. 8, No. 5, (2016), pp. 15-24, DOI: 10.5121/ijdms.2016.8502.

[4] Dahlan A & Wibowo FW, "Design of Library Data Warehouse Using SnowFlake Scheme Method: Case Study: Library Database of Campus XYZ", *Proceedings of IEEE 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, (2016), pp. 318-322, DOI: 10.1109/ISMS.2016.71.

[5] Ailamaki A, "Databases and hardware: the beginning and sequel of a beautiful friendship", *Proceedings of the VLDB Endowment*, Vol. 8, No. 12, (2015), pp. 2058-2061.

[6] Wibowo FW, "Interoperability of reconfiguring system on FPGA using a design entry of hardware description language", *Proceedings of the 2011 computation and communication technologies: 3rd international conference on advances in computing, control, and telecommunication technologies, ACT 2011 – Computer Science Series 1*, (2011), pp. 79-83.

[7] Papaphilippou, Philippos and Wayne Luk. "Accelerating database systems using FPGAs: A survey." (2018).

[8] Wibowo FW, Implementation of viterbi algorithm based-on field programmable gate array for wireless sensor network. *Advanced Science Letters*, Vol. 21, Issue 11, (2015), pp. 3521-3525.

[9] Sukhwani B, Min H, Thoennes M, Dube P, Brezzo BIB, Dillenberger D & Asaad S, "Database analytics acceleration using FPGAs", *Proceedings of the 21st international conference on Parallel architectures and compilation techniques*, (2012), pp 411-420, http://dx.doi.org/10.1145/2370816.2370874.

[10] Mahajan D, Kim JK, Sacks J, Ardalan A, Kumar A & Esmaeilzadeh H, "In-RDBMS hardware acceleration of advanced analytics", *Proceedings of the VLDB Endowment*, Vol. 11, No. 11, (2018), pp. 1317-1331, http://doi.org/10.14778/3236187.3236188.

[11] Wibowo FW, Sudarmawan & Sulistiyono M (2018), Hardware platform design analysis of k-means clustering algorithm implementation, *International Journal of Engineering & Technology*.