TELKOMNIKA, Vol. 11, No. 7, July 2013, pp. 3656 ~ 3664
e-ISSN: 2087-278X

Continuous Attributes Discretization Algorithm based on FPGA

Guoqiang Sun*[1,2], Hongli Wang[2] , Xing He[2], Jinghui Lu[2]
[1]Aviation University of Air Force, [2]The Second Artillery Engineering University
Number 2222 of Nanhu road, Changchun,130022,China, 0431-86958315
*Corresponding author, e-mail: guoqiangsun51@gmail.com

Abstract

The paper addresses the problem of Discretization of continuous attributes in rough set. Discretization of continuous attributes is an important part of rough set theory because most of data that we usually gain are continuous data. In order to improve processing speed of discretization, we propose a FPGA-based discretization algorithm of continuous attributes making use of the speed advantage of FPGA. Combined attributes dependency degree of rough ret, the discretization system was divided into eight modules according to block design. This method can save much time of pretreatment in rough set and improve operation efficiency. Extensive experiments on a certain fighter fault diagnosis validate the effectiveness of the algorithm.

***Keywords**: FPGA, rough set, dependency degree, discretization, fighter*

1. Introduction

Discretization of continuous attributes [1-2] is an important part of rough set theory. With improvement of the complexity of current systems and equipments, large amounts of fuzzy and uncertain data appear in the diagnosis process. These data are essentially continuous and incomplete, which bring difficulties for knowledge discovery and rule extraction.

Discretization of continuous attributes is essentially to set a number of discrete points in a specific range within the division, the scope of the attribute value is divided into a number of discretization intervals. At present, there are many discretization methods of continuous attributes in rough set, including the method of equal width and equal frequency [3], discretization algorithm base on information entropy [4], discretization algorithm based on attribute importance [5] and so on. Different methods produce different results, but any kind of discretization methods should meet the following two points as much as possible: Firstly, the space dimension should lack as much as possible after attribute discretization. Secondly, the number of information missing should lack as much as possible after attribute values are discreted.

Field programmable gate array (FPGA) [6-8] is the result of very large scale integration (VLSI) technology and computer-aided design (CAD) technology development. It adopts Logic Cell Array (LCA). FPGA is a kind of programmable logic device, which gained rapidly development in 1990's, after nearly twenty years of development, FPGA has become one of mainly platforms for realizing designing of digital system. Since the development of Moore's Law, we can buy FPGA of a few million gates and up to a few megahertz hertz frequency and can complete the designing of complex systems.

In this paper, we make use of speed advantage of FPGA and combination of attributes dependency degree in rough set theory, proposed a FPGA-based discretization algorithm of continuous attributes. Extensive tests are carried out to validate the proposed algorithm.

2. Rough Set Theory

Rough set theory [9] is a kind of mathematical methods, which is proposed by Z. Pawlak in 1982. This theory is mainly used for dealing with uncompleted and uncertain

Received January 24, 2013; Revised April 8, 2013; Accepted April 18, 2013

information. In control theory, knowledge discovery, decision support and analysis and fault diagnosis have received a wide range of applications.

Definition 1: Quadruple $DT = (U, C \bigcup D, V, f)$ is a decision table, U is non-empty finite set of objects, called the universe; C is condition attribute set; D is decision attribute set; V is the range of information function; f is information function of decision table.

Definition 2: If $C, D \subset A$ are two attribute sets, definition

$$\gamma(C,D) = k = \frac{\left|\sum_{X \in U/D} \underline{C}(X)\right|}{|U|}$$

is dependency degree between attribute C and attribute D, denoted by $C \Rightarrow_k D \left|\sum_{X \in U / D} \underline{C}(X)\right|$.is C positive region of D, it is an object collection that all information based on classification U / C in universe U can be divided accurately into equivalence class of relationship D.

Discretization algorithm of continuous attribute in rough set based on attribute dependency degree as follows:

1) Decision table $S = (U, C \bigcup D, V, f)$, to each continuous condition attribute $C_1 \in C$, Sort on the property value: $l_{C_1} = v_0^{C_1} < v_1^{C_1} < \cdots < v_n^{C_1} = r_{C_1}$, so you can select the candidate breakpoints as follows

$$C_i^{C_1} = (v_{i-1}^{C_1} + v_i^{C_1}) / 2 \text{ , } i = 1,2,\cdots,n$$

2) According to the formula of definition 2, calculate dependency degree of condition attributes, this formula can be extended to solve dependency degree that the decision attribute depends on a certain condition attribute;

3) Breakpoint of candidate breakpoint set in turn removed to form new breakpoint set, then make use of new breakpoint set to divide property value of the original property, and replaced by the interval and solve attribute dependency degree after divided;

4) Compared dependency degree by the third with dependency degree by the second, if dependency degree equal, did not produce new incompatibility, this breakpoint can be removed; if dependency degree did not equal, produce new incompatibility, this breakpoint must preserve;

5) Eventually the set in consisting of being preserved breakpoints as the last breakpoint set that can divide attribute value of the attribute.

Select a decision table about frost resistance of concrete. As table 1 shown, condition attribute set is $C = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}$,their value represents five physical properties of test result; the last attribute {Class} is decision attribute, the value 1 means antifreezing, the value 0 means non-antifreezing.

Table 1. A Decision Table About Frost Resistance of Concrete

U	α1	α2	α3	α4	α5	Class
1	0.007	1.90	2.90	0.80	3.40	1
2	0.004	1.15	2.95	4.10	4.00	1
3	0.004	1.15	6.15	0.80	0.350	1
4	0.008	1.95	2.85	1.30	0.450	1
5	0.008	2.60	2.85	1.30	0.350	1
6	0.006	1.15	6.15	0.90	3.40	1
7	0.005	2.60	6.10	0.85	3.45	1
8	0.007	1.30	4.20	0.85	3.45	1
9	0.009	2.65	6.10	2.10	4.00	0
10	0.048	2.60	6.00	2.15	0.350	0
11	0.016	3.90	1.20	2.30	4.00	0
12	0.019	4.00	4.20	2.65	4.00	0
13	0.035	2.65	1.20	4.10	4.50	0
14	0.008	2.65	4.20	1.30	0.450	0
15	0.016	2.75	1.35	1.35	0.450	0
16	0.007	1.85	6.15	1.30	4.35	0

(1) Select candidate breakpoint set $V'a$ of attribute α_1

$$V'a = \{0.0045, 0.0055, 0.0065, 0.0075, 0.0085, 0.0125, 0.0175, 0.0270, 0.0415\}$$

(2) Calculate dependency degree that decision attribute Class depends on condition attribute α_1

$$U/\alpha_1 = \{\{1,8,16\},\{2,3,7\},\{6\},\{9\},\{10\},\{11,15\},\{12\},\{13\}\}$$

$$U/Class = \{\{1,2,3,4,5,6,7,8\},\{9,10,11,12,13,14,15,16\}\}$$

Then dependency degree that decision attribute Class depends on condition attribute α_1 is

$$\gamma_{\alpha_1}(Class) = \frac{|POS_{\alpha_1}(Class)|}{|U|} = \frac{|\{2,3,6,7,9,10,11,12,13,15\}|}{16} = \frac{10}{16}$$

This formula show that there are ten objects can be classified accurately by property α_1 to equivalence class of $U/Class$.

(3) Start from attribute α_1, remove the first breakpoint 0.0045 of candidate breakpoint set, then make use of remaining breakpoints for dividing the original attribute value and expressed with a range or clarity. Then calculate dependency degree of attribute α_1 as follows

$$U/\alpha_{1new} = \{\{1,8,16\},\{2,3,7\},\{4,5,14\},\{6\},\{9\},\{10\},\{11,15\},\{12\},\{13\}\}$$

$$\gamma_{\alpha 1new}(Class) = \frac{|POS_{\alpha_1}(Class)|}{|U|} = \frac{|\{2,3,6,7,9,10,11,12,13,15\}|}{16} = \frac{10}{16}$$

The result equals to original dependency degree, the result shown that did not bring in new incompatibility, so breakpoint 0.0045 can be removed. If remove breakpoint 0.0065, then

$$U/\alpha_{1new} = \{\{1,8,16\},\{2,3\},\{4,5,9,14\},\{6\},\{7\},\{10\},\{11,15\},\{12\},\{13\}\}$$

$$\gamma_{\alpha 1new}(Class) = \frac{|POS_{\alpha_1}(Class)|}{|U|} = \frac{|\{2,3,6,7,10,11,12,13,15\}|}{16} = \frac{9}{16}$$

The result did not equal to original dependency degree, the result shown that bring in new incompatibility, so this breakpoint can not be removed and must preserve.

(4) Gain breakpoint set of attribute α_1 is $\{0.0065, 0.0085\}$, attribute value of the original attribute can be divided by these breakpoints and gain division interval, and make use of the entire value $\{0,1,2,\cdots\}$ to replace for each division interval, that is division result of original attribute as table 2 shown.

Table 2. Discretization Decision Table

U	α1	α2	α3	α4	α5	Class
1	1	1	1	0	2	1
2	0	0	1	4	3	1
3	0	0	3	0	0	1
4	1	1	1	1	0	1
5	1	2	1	1	1	1
6	0	0	3	0	2	1
7	0	2	3	0	2	1
8	1	0	2	0	2	1
9	2	2	3	2	3	0
10	2	2	3	2	0	0
11	2	3	0	2	1	0
12	2	3	2	3	3	0
13	2	2	0	4	3	0
14	1	2	2	1	1	0
15	2	2	0	1	1	0
16	1	1	3	1	3	0

3. Discretization Algorithm of Continuous Attributes Based on FPGA

In this section, we make use of the FPGA to realize discretization of continuous attributes based on attribute dependency degree, and according to the method, decision table of frost resistance of concrete is processed, simulated and verified.

3.1. Continuous Attribute Discretization System Designing

The system is divided eight modules, including the sorting module, the candidate breakpoint module, the fine selection breakpoint module, the separation breakpoint module, the based classification module, the separated classification module, the output module and the control module. The sort module is mainly sorting from small to large; the candidate breakpoint module mainly selects candidate breakpoints; the fine selection breakpoint module mainly resorts candidate breakpoints; separation breakpoint module mainly used to delete breakpoint of breakpoint set regularly; based classification module as a reference module mainly calculated dependency degree of the original attributes; separated classification module mainly used to calculate dependency degree that the attributes though dividing; output module mainly used to output the last result; control module mainly used to control the whole workflow. System structure as figure 1 shown.

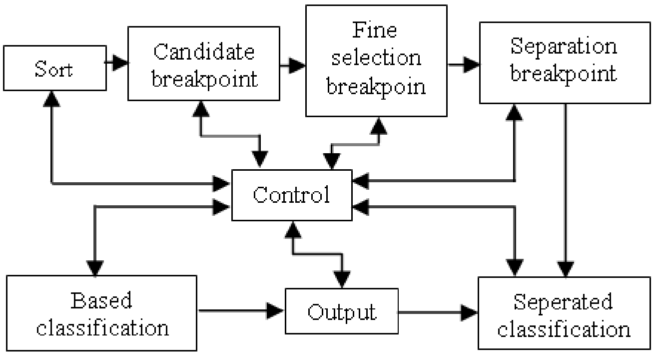

Figure 1. System structure figure

3.2. Running Process Of The System

(1) When the FPGA development board effective after the power or reset signal, the signal of each module is initialized;

(2) After the initialization, control module brings in control signal, at the same time, sort module began to work;

(3) After sorting, selected candidate breakpoints, principle is that the same does not mean, not the same mean, for example, 40,40,60,70 candidate breakpoints are 50 and 65;

(4) The separation breakpoint module mainly supplies breakpoints for separated classification module, divided attribute value of the original attribute, then calculates dependency degree that decision attribute depended on a certain condition attribute;

(5) The dependency degree obtained from the based classification module called the original dependency degree, the dependency degree obtained from the separated classification module called the current dependency degree, if the current dependency degree equals to the original dependency degree, show that did not bring in new incompatible, this breakpoint is able to be deleted, otherwise reserved. we can find the set of breakpoints through this method;

(6) According to the obtained breakpoints, original decision table attribute value will be divided, then use the integer value for replacing the interval as the result of discretization of continuous attribute.

3.3. Important Module Simulation And System Simulation

Figure 2 is the basic classification module structure figure.

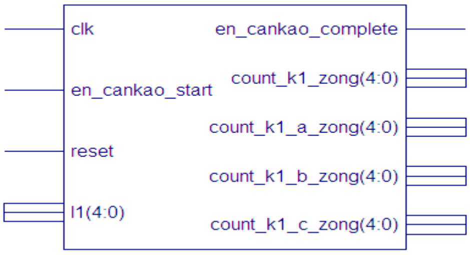

Figure 2. Basic classification module structure figure

Where, clk is clock signal, reset is initialization signal, en_cankao_start is enable signal, $I1(4:0)$ is each attribute of control attribute set, en_cankao_complete is completion signal, count_k1_a_zong and count_k1_b_zong and count_k1_c_zong are numbers of correct classification obtained though different decision attribute values, count_k1_zong is the total numbers of correct classification by calculation.

Figure 3 is the basic classification module simulation figure of attribute a. As can be seen from the simulation figure, when en_cankao_complete equals to 1, there are ten objects that can be classified accurately according to attribute a, that is, count_k_zong equals to 10.

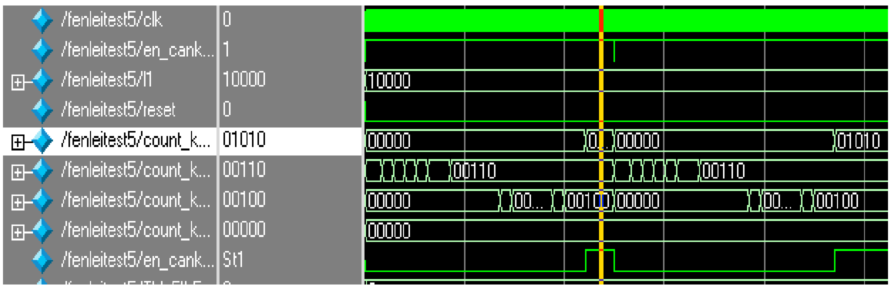

Figure 3. Basic classification module simulation figure

Figure 4 is separated classification module structure figure. $q1, q2, \cdots, q14$ are outputs of separation breakpoint module, en_fenbie_start is enable signal of separated classification module, en_fenbie_a, en_fenbie_b, en_fenbie_c, en_fenbie_d, en_fenbie_e are enable signal of controlling each attribute, count_a_zong, count_b_zong, count_c_zong are numbers of correct classification obtained though different decision attribute values, count_zong=count_a_zong+ count_b_zong+ count_c_zong are the total numbers of correct classification by calculation.

Figure 5 is separated classification module simulation figure after removing 0.0045 and 0.0065. As can be seen from the simulation figure, when en_fenbie_complete equals to 1, there are 10 objects that can be classified accurately according to attribute a after being divided by breakpoint set, that is, count_zong equals to 10; when en_fenbie_complete equals to 1 again, there are 9 objects that can be classified accurately according to attribute a after being divided by breakpoint set, that is, count_zong equals to 9; Here can illustrate, objects that are classified accurately change, and bring about a new incompatible.

Figure 6 is output module structure figure. Where, clk is clock signal, reset is initialization signal, en_shuchu_start is enable signal of output module, $o1, o2, \cdots, o15$ are outputs of fine selection breakpoint module, en_shuchu_a, en_shuchu_b, en_shuchu_c, en_shuchu_d, en_shuchu_e are enable signal of each attribute, ji_chu is the value of count_k1_zong, fen_bie is the the value of count_zong, count is the number of calculating the candidate breakpoint, en_wancheng is the final result of the output module, the function of the en_jixu is relative to count, *Out1, out2, … out16* are results of the output.

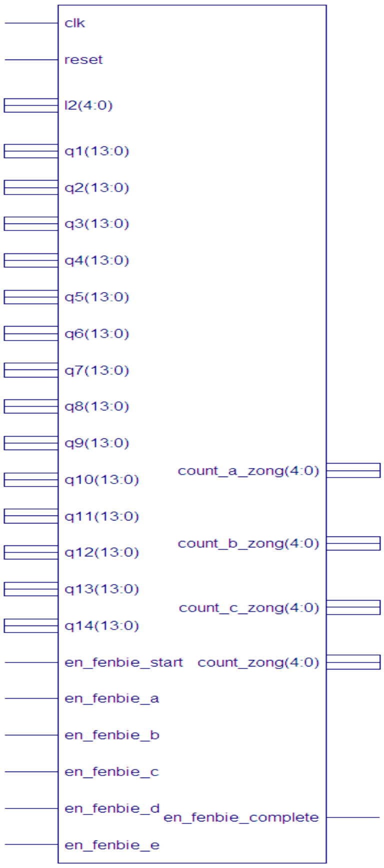

Figure 4. Separated classification module structure figure

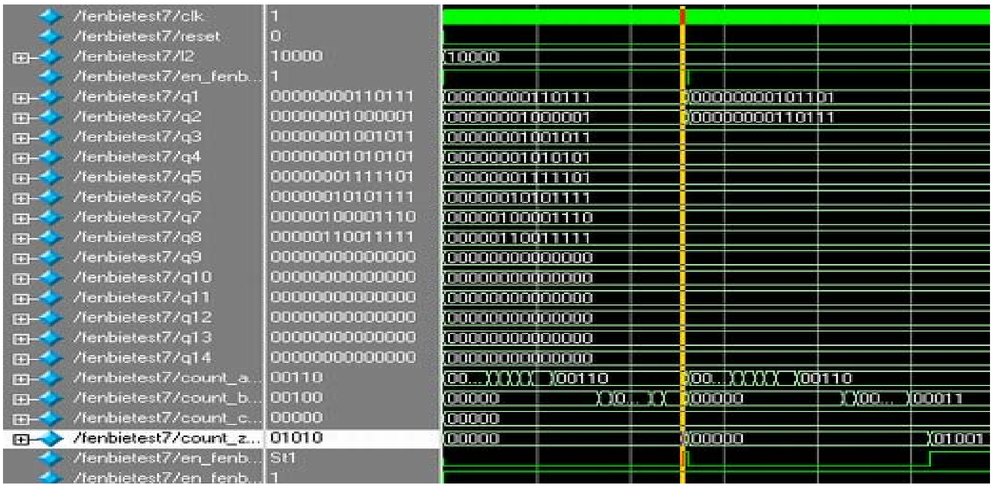

Figure 5. Separated classification module simulation figure

Figure 7 is discretization simulation figure of attribute a: As can be seen from simulation figure. After comparing, we can ensure breakpoints that must be retained finally, then gain discretization result of original attribute a.

Figure 8 is simulation figure of discretization system of continuous attributes. *Out1, out2, … out16* showed the last result. The whole discretization process takes approximately 38.5ms.The result uses binary for expressing, for example, binary 00000 expressed integer zero, binary 00011 expressed integer three.

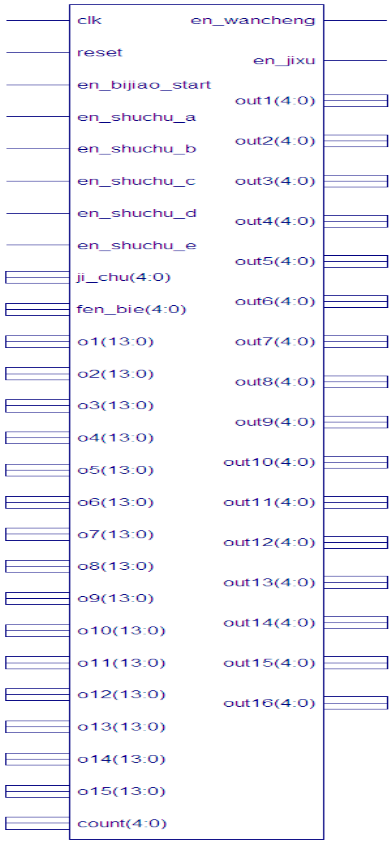

Figure 6. Output module structure figure

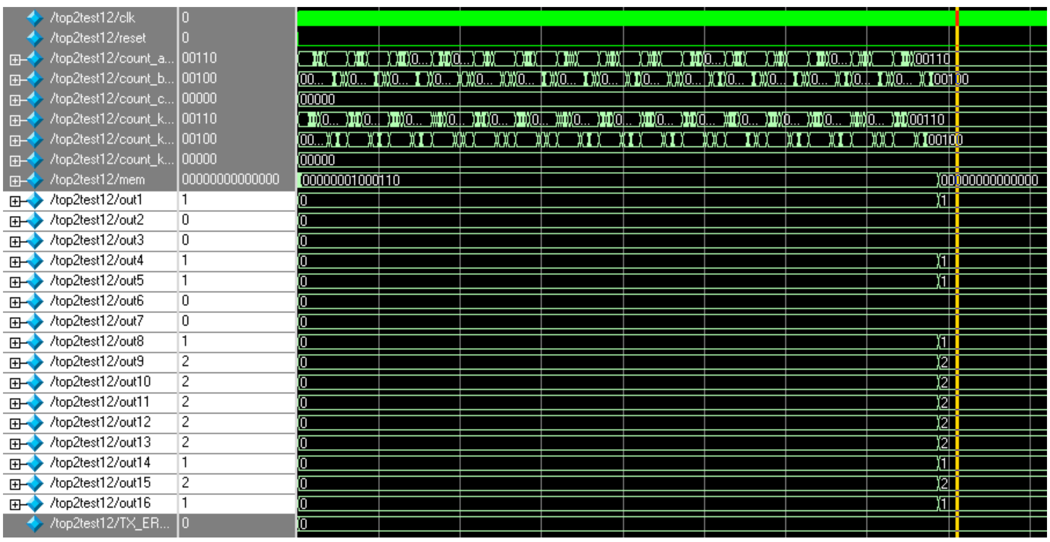

Figure 7. Discretization simulation figure of attribute a

4. Evaluations on Fighter Diagnosis

The data used come from the output of a nonlinear aircraft model. The flight altitude is 5,000 meters, the flight Mach number is 0.6, sampling time is 0.012 seconds. Collected the following sample data about 223 sets: the normal state data, the horizontal tail stick data of -5ºand -10º, the horizontal tail injury data of 20%, 50%, 100%, the aileron stick data of -20º, -10º, 10º, 20º, the aileron injury data of 20% ,40%, 80%, 100%.

Figure 8. The whole system simulation figure

Each sample data include eight variables: attacking angle (Alpha), sideslip angle (Beta), rolling velocity (Wx), pitch rate (Wz), yaw rate (Wy), yaw angle (Psi), pitch angle (Theta) and roll angle (Gamma). The decision table as Table 3 shown. Adopting the purposed method, the discretization result of table 3 as Table 4 shown.

Table 3. Decision Table

U	Alpha	Beta	Wx	Wz	Wy	Psi	Theta	Gamma	D
1	5.0692	0	0	-0.0101	0	5.0692	5.0697	0	1
…	…	…	…	…	…	…	…	…	…
27	5.0666	0	0	-0.0137	0	5.0666	5.0673	0	1
28	5.4110	0	18.0000	2.0017	-1.0000	5.4110	5.4010	3.0000	2
…	…	…	…	…	…	…	…	…	…
57	7.5899	5.0000	81.0000	3.8100	-7.0000	7.5899	6.4733	78.0000	2
58	5.0564	0.0001	12.0000	0.0510	-1.0000	5.0564	5.0396	6.0000	3
…	…	…	…	…	…	…	…	…	…
98	5.0438	0.0000	13.0000	0.0708	-1.0000	5.0438	4.7439	23.0000	3
99	4.9550	0.0006	48.0000	0.0372	-3.0000	4.9550	4.9230	13.0000	4
…	…	…	…	…	…	…	…	…	…
163	4.4068	2.0000	42.0000	0.7729	-2.0000	4.4068	3.8219	56.0000	4
164	4.4111	-2.0000	-46.0000	0.4188	1.0000	4.4111	4.8342	-34.0000	5
…	…	…	…	…	…	…	…	…	…
223	5.0211	1.0000	7.0000	0.1218	-1.0000	5.0211	4.5508	31.0000	5

Table 4. Discrete Decision Table

U	Alpha	Beta	Wx	Wz	Wy	Psi	Theta	Gamma	D
1	3	0	0	0	0	2	3	0	1
…	…	…	…	…	…	…	…	…	…
27	3	0	0	0	0	2	3	0	1
28	7	0	2	1	1	4	3	1	2
…	…	…	…	…	…	…	…	…	…
57	3	0	0	0	0	2	3	0	2
58	3	0	0	0	0	2	2	0	3
…	…	…	…	…	…	…	…	…	…
98	3	0	1	0	0	2	2	0	3
99	3	0	3	0	1	2	2	0	4
…	…	…	…	…	…	…	…	…	…
163	0	0	3	2	0	0	2	1	4
164	3	0	0	0	0	2	3	0	5
…	…	…	…	…	…	…	…	…	…
223	3	0	0	0	0	2	2	0	5

System simulation figure of discretization as Figure 9 shown.

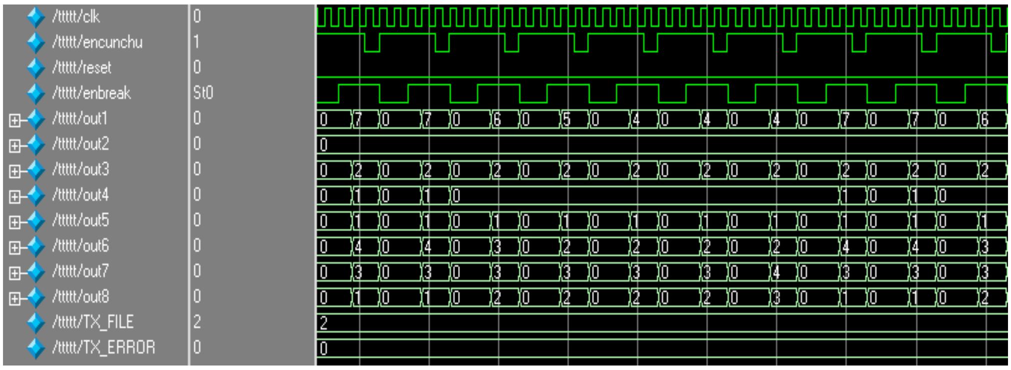

Figure 9. System simulation figure of discretization

5. Conclusion

Rough set theory is mainly used for processing fuzzy and uncertain information and knowledge. Discretization of continuous attributes of rough set theory is used to pretreat and ensure that the discretization result is simple and consistent. At present, there are many discretization methods, but most of them are based on software.

In this paper, we make use of the speed advantage of hardware to design discretization system which was divided into eight modules. This algorithm is able to ensure that the compatibility of decision tables is not changed after discretization. Meanwhile, this method can save much time of pretreatment and improve operation efficiency. Extensive experiments on a certain fighter fault diagnosis validate the effectiveness of the algorithm.

References

[1] Liping An. *Multiple Attribute Decision Analysis Based On Rough Set*. Theory And Science Press, BeiJing. 2008.
[2] C Shousong Hu. *Decision Theory And Application Based On Rough Set*. Beijing Aeronautics and Astronautics Press, BeiJing. 2006.
[3] Yilun cheng. Research on Discretized Data Methods Based on Rough Set Theory. Master Thesis of Jilin University. 2009.
[4] Yonghong shen. Method of attributes discretization in rough sets using information entropy and its application. *Computer Engineering and Applications*. 2007; 10: 221-223.
[5] Lingxia Liu. Discretization algorithm based on attribute importance in rough set theory. *Computer and Information Technology*. 2007; 44: 221-223.
[6] Cox CE and E Blanz, GangLion. A Fast Field-programmable Gate Array Implementation of A Connectionist Classifier. *IEEE Journal of Solid-State Circuits*.1992; 28: 288-299.
[7] Tian Yun, Wenbo Xu, and Hu Bin. Xinlinx ISE Design Suite 10.X FPGA Developer's Guide-Logic Design Posts, Posts & Telecom Press, BeiJing. 2008.
[8] Zhijun Xu, Guanghui Xu. Development And Application Based On CPLD And FPGA. BeiJing. 2002.
[9] Pawlak Z. Rough sets. *International J of Computer and Information Sciences*. 1982, 11: 341-356.