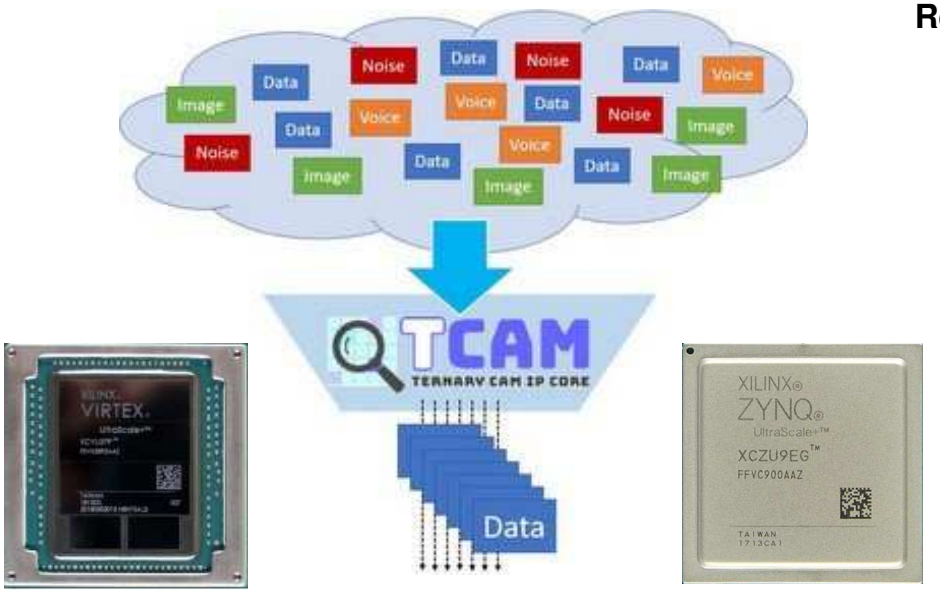
tCAM-IP for Xilinx introduction

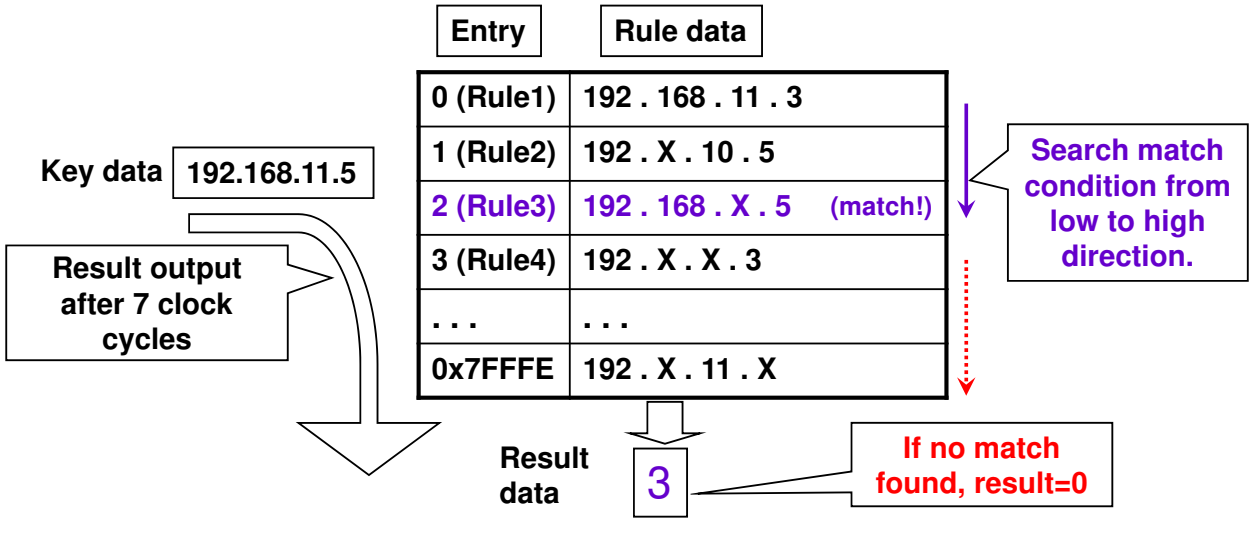
Rev1.0XE



Super-low latency Ternary-CAM IP-Core

tCAM-IP core summary

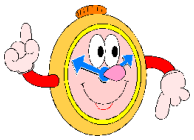
- Super low latency ternary (0,1,X) CAM controller IP core.
- Searching latency=7 clock (constant), up to 512K rule entry.



tCAM-IP core operation image

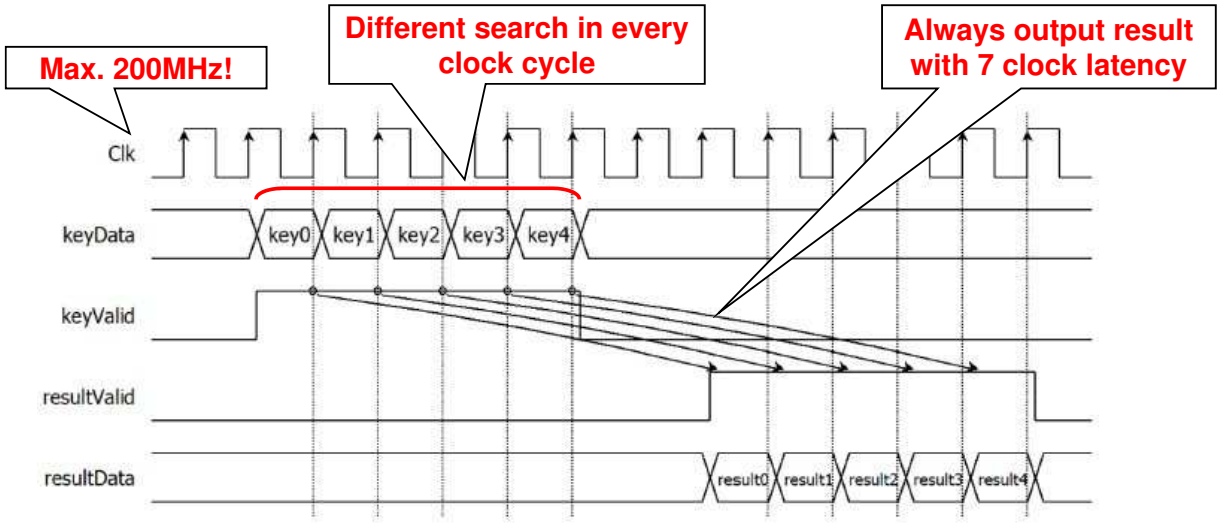
tCAM-IP core advantage

- Continuous operation under 200MHz clock speed.
- Searching latency is constant at 7 clock cycles.
- Up to 512K rule entries.
- Key bit width 64/56/48/40/32/24 bits
- Support customization
  - Expand rule entry count or key bit width.
  - Support external memory usage.



tCAM-IP core advantage (detail1)

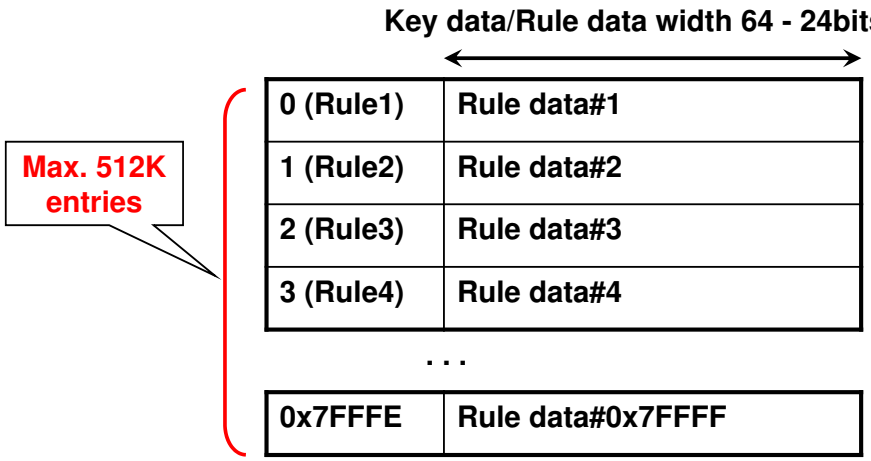
- Searching latency is constant at 7 clock cycles.
- Continuous search operation in every clock (fmax=200MHz).



Timing waveform during search operation

tCAM-IP core advantage (detail2)

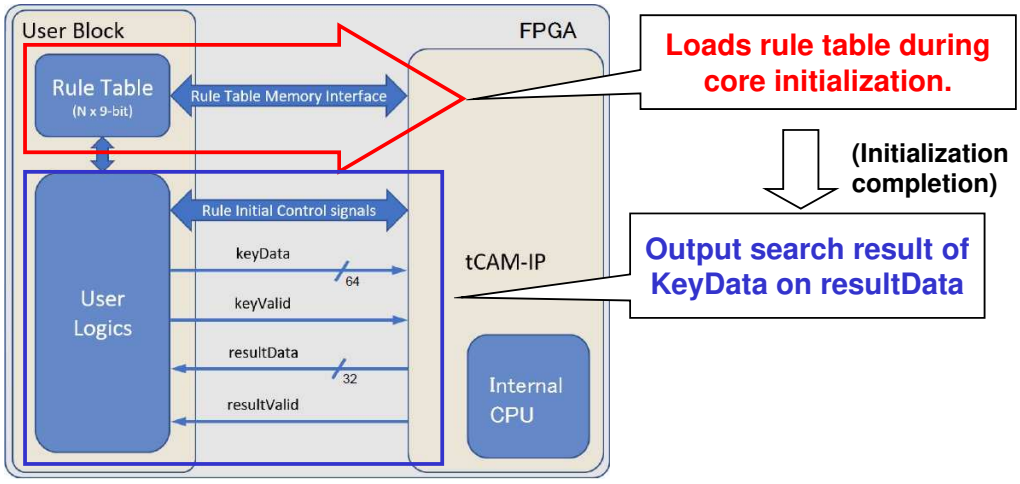
- **Rule count 512K entries at maximum.**
- **Key data width: Selectable from 64/56/48/40/32/24 bits.**



tCAM-IP rule table

tCAM-IP core operation

- **tCAM-IP core loads rule table during initialization.**
- **Search result of KeyData appears on resultData.**



tCAM-IP core operation

tCAM-IP core initialization

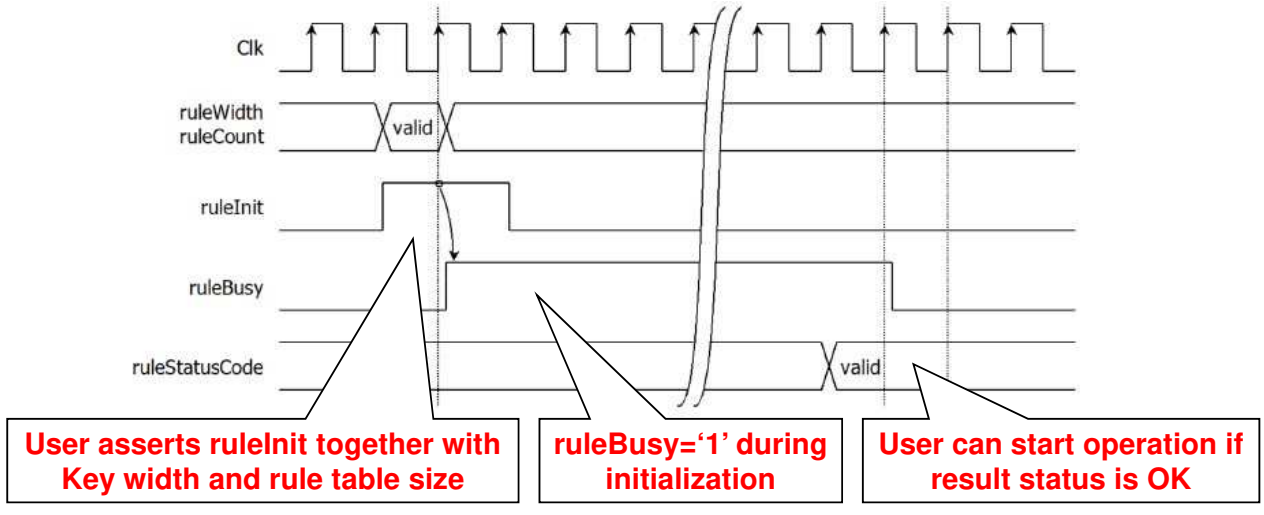
- Initialization starts by ruleInit assertion.
- tCAM-IP core loads rule table from user block area.
- tCAM-IP negates ruleBusy after initialization.
- If result status is OK, search operation is ready.
- Initialization time depends on key and rule table size.

Key width	Rule size	Init. time
32bit	1K	523msec
32bit	64K	2,957msec
64bit	1K	620msec
64bit	64K	15,472msec

tCAM-IP core initialization time

tCAM-IP core initialization (detail1)

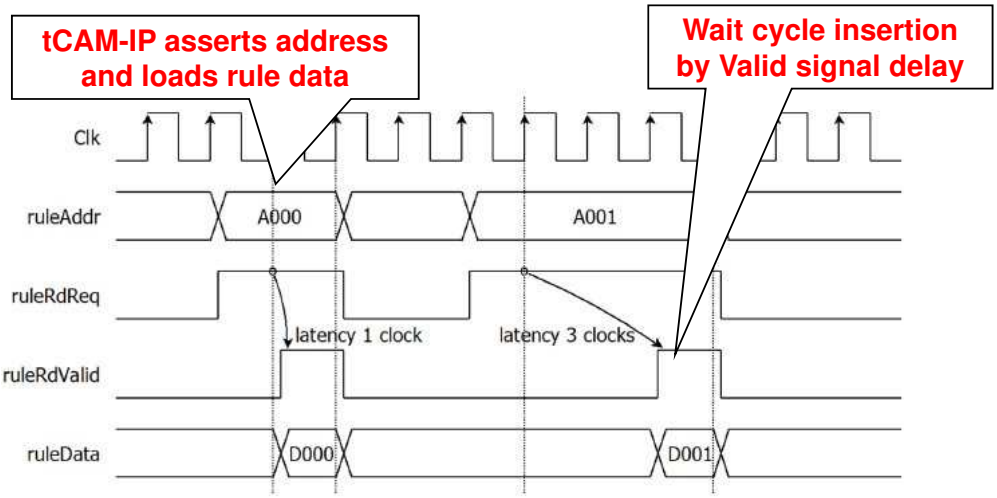
- tCAM-IP loads rule data from user block area



tCAM-IP initialization timing waveform

tCAM-IP core initialization (detail2)

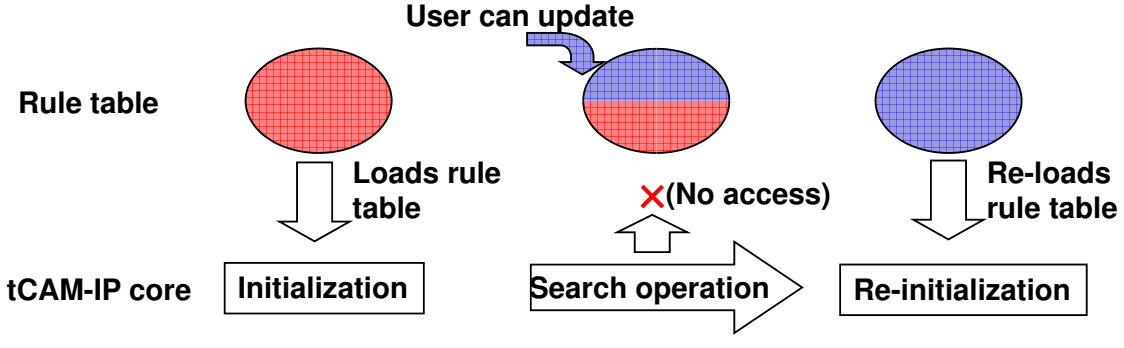
- tCAM-IP loads and scans rule table data in user block.
- User can insert wait cycle by controlling ruleRdValid.



tCAM-IP core rule data access timing waveform

Rule table update

- tCAM-IP does not access to rule table after initialization.
- So user can update rule data during core operation.
- tCAM-IP re-loads updated data by re-initialization.



Rule table update during search operation

Resource usage

- **Resource usage of FPGA fabric and block memory**

Example Implementation Statistics

Family	Example Device	Fmax (MHz)	CLB Regs	CLB LUTs	CLB[1]	IOB	BRAMTile[2]	Design Tools
Kintex UltraScale+	XCKU5P-FFVB676-2-E	200	3749	3344	1519	-	292	Vivado2019.1

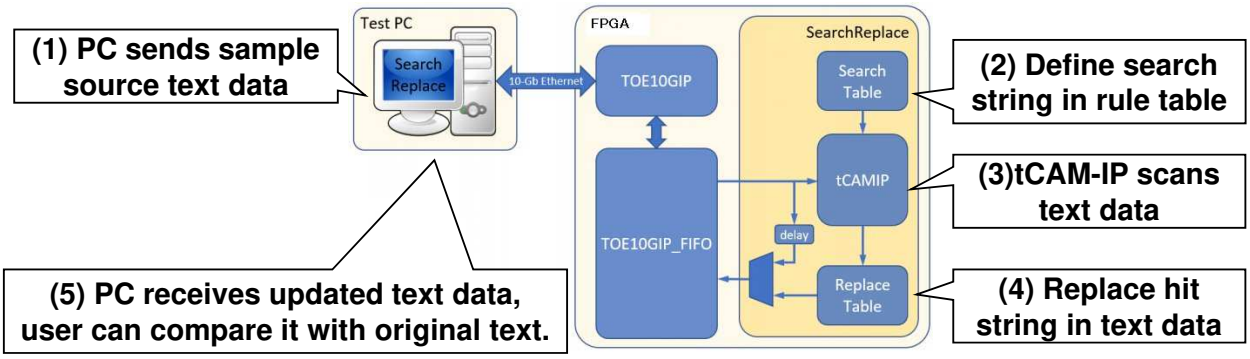
Notes:

1) Actual logic resource dependent on percentage of unrelated logic
2) Exclude user rule table memory, Ex: 256K x 9-bit rule table memory will take 64 BRAMTile.

tCAM-IP core resource usage

Demo1 (Search&Replace demo)

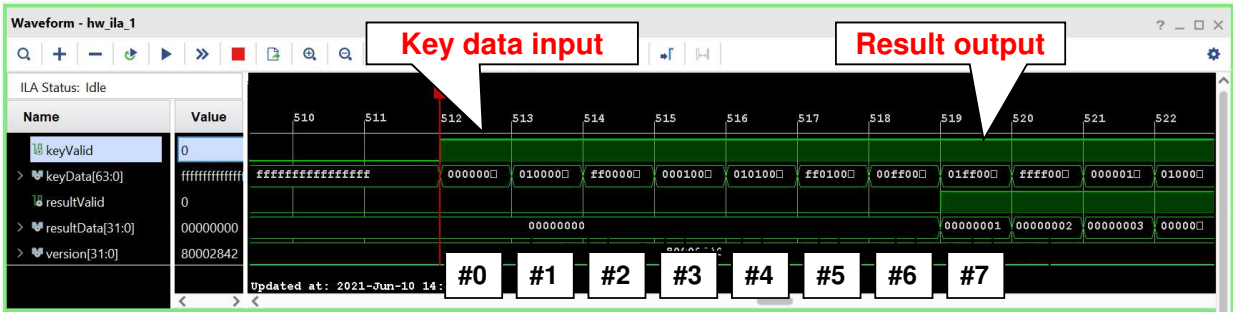
- **Connect PC and KCU116 by 10GbE.**
- **PC sends sample source text data to FPGA.**
- **tCAM-IP core searches specified string in text data, then replace it with another string.**
- **PC receives updated text data after replacement.**



tCAM-IP core search and replace demo

Demo2 (ChipScope waveform)

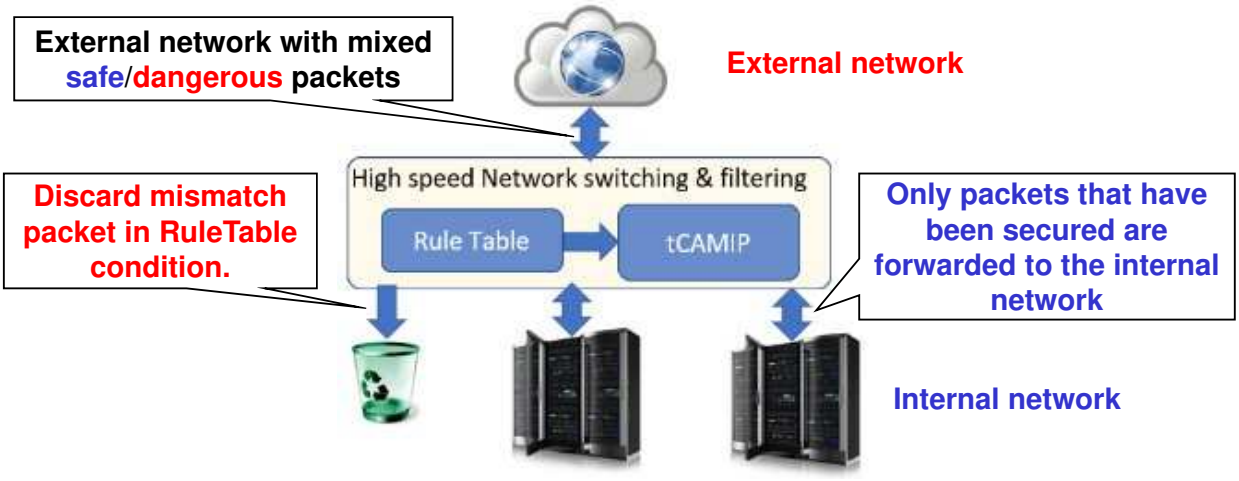
- Measure real search time by ChipScope.
- System clock frequency = 200MHz.
- User can confirm 7 clocks latency search time.



tCAM-IP waveform measured by ChipScope

Application example

- Network switch or filtering system.



Network security system by tCAM-IP

For more detail

- **Detailed technical information available on the web site.**
  https://dgway.com/tCAM-IP_X_E.html
- **Contact**
  - **Design Gateway Co,. Ltd.**
  - **sales@design-gateway.com**
  - **FAX: +66-2-261-2290**

Revision History

Rev.	Date	Description
1.0XE	5-Aug-21	1st English Revision (Rev1.0XE)