

KNN solver with hardware acceleration

IOB-KNN User Guide, V0.1 , Build 5e7fb2f



January 2, 2021



Contents

1	Introduction	5
2	Symbol	5
3	Features	5
4	Benefits	6
5	Deliverables	6
6	Block Diagram and Description	7
7	Synthesis Parameters	8
8	Interface Signals	8
9	Registers	9
10	FPGA Results	10

List of Tables

1	Block descriptions.	7
2	General Interface Signals	8
3	CPU Native Slave Interface Signals	8
4	CPU AXI4 Lite Slave Interface Signals	9
5	Software accessible registers.	9
6	Implementation Resources for Xilinx Kintex Ultrascale Devices	10

List of Figures

1	IP Core Symbol	5
---	--------------------------	---



2 High-level block diagram 7

1 Introduction

The IObundle KNN core is an hardware KNN algorithm solver that attributes the labels to a maximum of 10 labels. It is written in Verilog and includes a C software driver. The IP is currently supported for use only in FPGAs.

2 Symbol

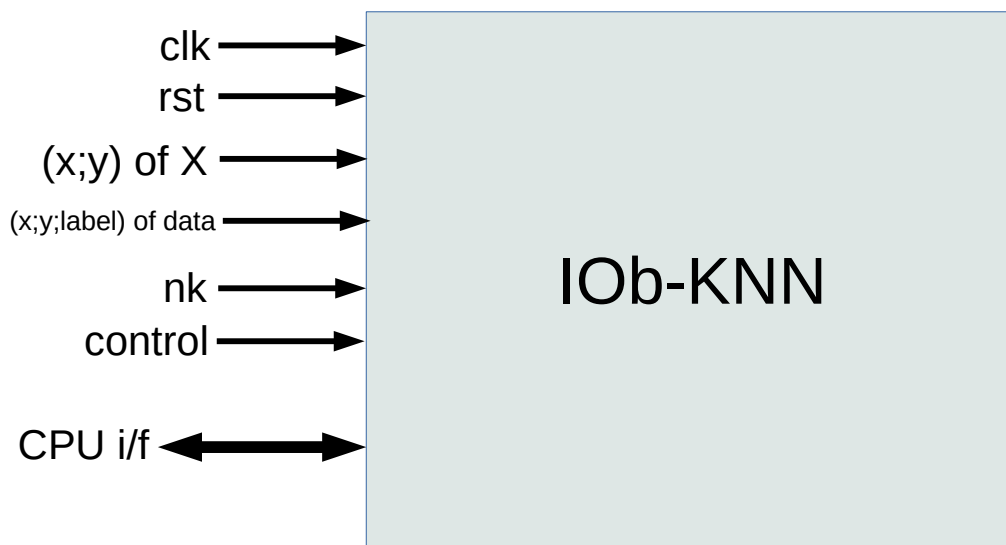


Figure 1: IP Core Symbol

3 Features

- Verilog KNN solver accelerator (16 bit operands);
- C software driver;
- Reset, enable and time read functions;
- IOb-SoC native CPU interface.

4 Benefits

- Compact hardware implementation;
- Can fit many instances in low cost FPGAs;
- Low power consumption;
- Quick resolution of KNN algorithm;
- Parallel design.

5 Deliverables

- Verilog source code;
- User documentation for easy system integration;
- Example integration in IOB-SoC;
- FPGA synthesis and implementation scripts.

6 Block Diagram and Description

A high-level block diagram of the IOB-KNN core is presented in Figure 6 and a brief explanation of each block is given in Table 1.

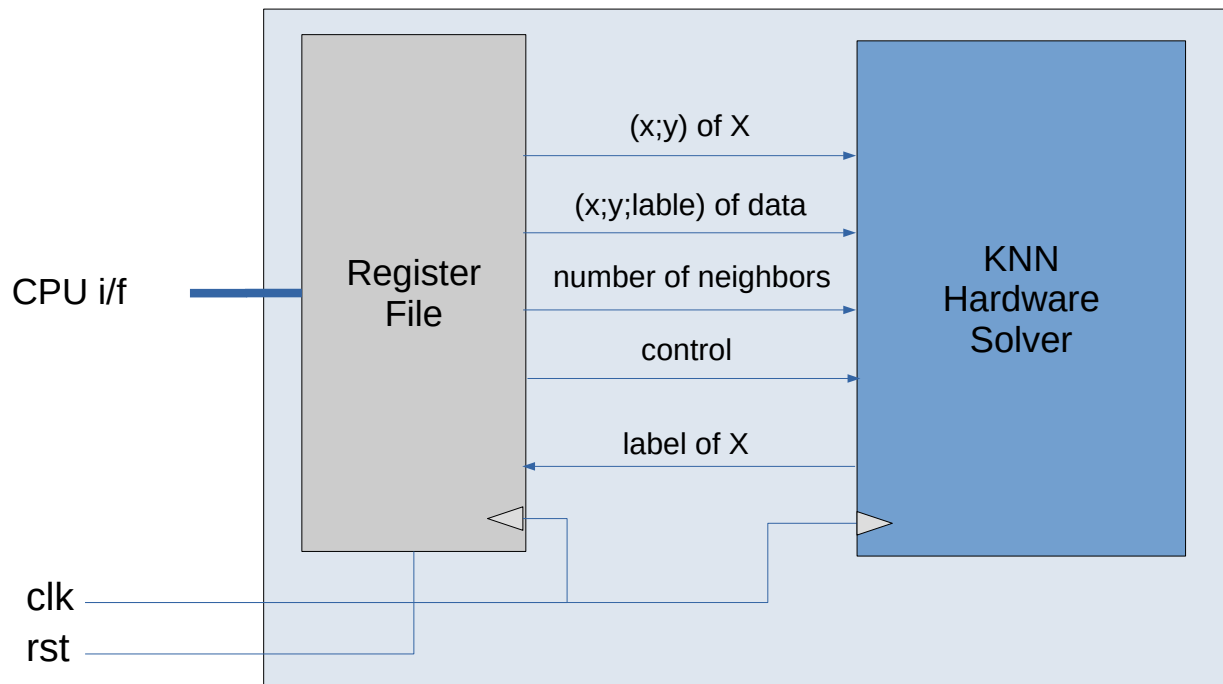


Figure 2: High-level block diagram

Block	Description
Register File	Configuration, control and status registers accessible by the software
KNN	

Table 1: Block descriptions.

7 Synthesis Parameters

The KNN module is independent of the number of point, labels (maximum of 10) and neighbors (max of 10) and therefore the synthesis of the module is constant.

8 Interface Signals

The interface signals of the I²S/TDM transceiver core are described in the following tables.

Name	Direction	Width	Description
clk	input	1	System clock input
rst	input	1	System reset asynchronous and active high

Table 2: General Interface Signals

Name	Direction	Width	Description
valid	input	1	Native CPU interface valid signal
address	input	ADDR_W	Native CPU interface address signal
wdata	input	WDATA_W	Native CPU interface data write signal
wstrb	input	DATA_W/8	Native CPU interface write strobe signal
rdata	output	DATA_W	Native CPU interface read data signal
ready	output	1	Native CPU interface ready signal

Table 3: CPU Native Slave Interface Signals

Name	Direction	Width	Description
s_axil_awaddr	input	ADDR_W	Address write channel address
s_axil_awcache	input	4	Address write channel memory type. Transactions set with Normal Non-cacheable Modifiable and Bufferable (0011).
s_axil_awprot	input	3	Address write channel protection type. Transactions set with Normal Secure and Data attributes (000).
s_axil_awvalid	input	1	Address write channel valid
s_axil_awready	output	1	Address write channel ready
s_axil_wdata	input	DATA_W	Write channel data
s_axil_wstrb	input	DATA_W/8	Write channel write strobe
s_axil_wvalid	input	1	Write channel valid
s_axil_wready	output	1	Write channel ready
s_axil_bresp	output	2	Write response channel response
s_axil_bvalid	output	1	Write response channel valid
s_axil_bready	input	1	Write response channel ready
s_axil_araddr	input	ADDR_W	Address read channel address
s_axil_arcache	input	4	Address read channel memory type. Transactions set with Normal Non-cacheable Modifiable and Bufferable (0011).
s_axil_arprot	input	3	Address read channel protection type. Transactions set with Normal Secure and Data attributes (000).
s_axil_arvalid	input	1	Address read channel valid
s_axil_arready	output	1	Address read channel ready
s_axil_rdata	output	DATA_W	Read channel data
s_axil_rresp	output	2	Read channel response
s_axil_rvalid	output	1	Read channel valid
s_axil_rready	input	1	Read channel ready

Table 4: CPU AXI4 Lite Slave Interface Signals

9 Registers

The software accessible registers of the KNN core are described in Table 5. The table gives information on the name, read/write capability, word aligned addresses, used word bits and a textual description.

Name	R/W	Addr	Bits	Initial Value	Description
NK	W	0x00	7:0	0	number of neighbors
XX	W	0x04	WDATA_W-1:0	0	x coordinate for point being studied
YY	W	0x08	WDATA_W-1:0	0	y coordinate for point being studied
DATA_X	W	0x0c	WDATA_W-1:0	0	x coordinate for data point
DATA_Y	W	0x10	WDATA_W-1:0	0	y coordinate for data point
DATA_LABEL	W	0x14	7:0	0	data label
CONTROL	W	0x18	3:0	0	KNN reset and control (LSB reset the others control)
XLABEL	R	0x1c	31:0	0	label of the studied point

Table 5: Software accessible registers.

10 FPGA Results

The following are FPGA implementation results for two FPGA device families.

Resource	Used
LUTs	1212
Registers	490 Registers
490	
DSPs	2
BRAM	0

Table 6: Implementation Resources for Xilinx Kintex Ultrascale Devices