T-COR-11 FPGA IP CORE FOR TRACKING OBJECTS IN VIDEO STREAM IMAGES

Programmer manual

IP core version: 1.1
Date: 28.09.2015

CONTENTS

INTRODUCTION

The T-COR-11 FPGA IP core for tracking objects in video stream images is designed for use in vision systems intended for various applications (security systems, specialized systems, etc.) that are based on FPGA. The core is a completed module suitable for use in any FPGA projects. The core represents a simple data exchange interface that ensures easy integration into various systems. The core makes use of a modified high-performance correlation tracking algorithm which allows tracking of all types of objects. The implemented algorithms provide stable tracking of small-size and low-contrast objects against a complex background. The algorithms are perfectly suitable for tracking ground, surface and aerial objects of any type. Capture of objects for tracking is carried out by a command. In case of loss of the object (tracking break), the algorithms predict the path followed by the object up to automatic re-capture. Thus, the T-COR-11 IP core is a versatile module that allows its use in any FPGA projects for vision systems.

CORE VERSIONS

Versions of the T-COR-11 IP core are shown in Table 1.

Table 1 – Versions of the T-COR-11 IP core.

Version	Notes
T-COR-10	First version of the IP core. Implements a modified high-performance correlation tracking algorithm.
T-COR-11	Improved the quality of tracking. Need in DDR memory for better compatibility is eliminated. Block memory is used for service data storage.

BASIC CHARACTERISTICS

The T-COR-11 IP core ensures tracking of objects in video sequence images. The basic characteristics of the core are given in Table 2.

Table 2 – Basic characteristics of the T-COR-11 IP core.

Parameter	Values and notes
Maximum and minimum sizes of tracking strobes	The maximum size of tracking strobe is 128x128 pixels. The minimum size of tracking strobe is 16x16 pixels. The strobe size can be set separately for each object, both before object capture for tracking and during tracking.
Number of tracking channels	5 independent tracking chanels.
Maximum size of tracking object	Maximum size of the tracking object is limited by maximum size of the tracking strobe (128x128 pixels).
Minimum size of tracking object	The core ensures stable tracking of objects of up to 8x8 pixels size.
Maximum movement speed of tracking object	For a sudden change of the position of tracking object, the speed is no more than 20 pixels per frame in any direction. For a steady movement of the tracking object, the speed is limited by the capability of the tracking system used (drives speed).
Minimum contrast of tracking objects	The core ensures stable tracking of objects featuring from to 10% contrast.
Maximum size of frames	The core operates with frames of up to 2048x2048 pixels size.
Elongation of tracking objects	Tracking objects can have any configuration within the maximum size of tracking strobes.
Output coordinates accuracy	Output coordinates of tracking objects are obtained with 1 pixel accuracy.
Time lag of output coordinates	At the frame refresh rate of 25 Hz and recommended sizes of the tracking strobe, the core calculates the object coordinates before the end of the next frame in the video sequence (delay is 1 frame).

Parameter	Values and notes
Format of processed images	The core works with images having color palette of 256 colors (grayscale) – 8 bits/pixel.
Maximum frame rate	Until for 30 Hz.

Note: the data above is given for FPGA Xilinx Kintes-7.

DESCRIPTION AND OPERATION PRINCIPLE

The core is a complete module intended for use in FPGA system vision projects. It works as follows. The core works according the following principle. The core is connected to the frame buffer controller (can be either provided by RIFTEK or designed by the customer) via simple data exchange interface. Frame buffer can store images into block memory or in DDR memory. The designer should ensure frame image is updated into frame buffer (the frame buffer controller provided by RIFTEK includes two frame buffers, new image frame is recorded into the first one while the second one works with the core, when the frame is over the buffers swap places). Before operation is started, the core should be configured (configuration register values need to be written). Capture of objects for tracking and rejection from tracking is carried out by a command. The core supports independent tracking of up to 5 objects. Sizes of the tracking strobes are set separately for each object and can be changed in the course of tracking. The core calculates the position of tracking objects for each new frame, and the output coordinate is determined with a delay equal to 1 frame. The output data of the core is the coordinates of the object tracked.

APPLICATION FIELD AND PLATFORMS USED

The core can be used in any vision applications such as security video surveillance systems and in special robotic vision systems. A simple interface of data exchange with the core makes it possible to integrate it easily into any FPGA project. The core is made available to customer with specification of FPGA type. The core is delivered to the customer in a synthesized form for a specific type of FPGA under license. The core was developed for FPGA's of Xilinx Company and supports the following of their families:

Artix-7, Artix-7Q;
Zenq-7, Zenq-7Q;
Kintex-7, Kintex-7Q;
Virtex-7, Virtex-7Q;
Virtex-6, Virtex-6Q;
Spartan-6;
Virtex-5Q, Virtex-5QV;
Virtex-4Q, Virtex-4QV.

PROCEDURE OF USE

The use of the core the designer should provide a frame buffer with the capacity to read a part of an image starting from any point and of any length. Data readout logic must provide their 4 bytes alignment depending on the 2 lower bits of the reading start address. Before starting the operation and every time the core is powered up, it is necessary to configure the device (write configuration registers). Once the configuration registers are written, it is possible to carry out control of tracking. The core does not provide a display of information. The display should be provided by the programmer. A detailed description of the core interface and the procedure of use are given in next parts of this document.

PERFOMANCE

The IP core characteristics given in Table 3 have been calculated from the results of tests performed on FPGA Xilinx Kintex-7. The performance characteristics depend significantly on the type of FPGA and image refresh rate. Table 3 shows throughput characteristics of the core and their specific features.

Table 3 – Basic characteristics of the T-COR-11 IP core and their specific features.

Parameter	Values and notes
Tracking mode	Correlational. Implemented in the core is a modified high-performance correlation algorithm of tracking. Algorithms are employed that allow extrapolation of trajectory of tracking objects and control of tracking break.
Maximum number of objects tracked simultaneously	The number of tracking objects depends on the size of tracking strobe and FPGA microcircuit used. For FPGA Xilinx Kintex-7 and frame refresh rate of 25 Hz, the following relationships are true: strobe size 40x40 pixels – 5 objects; strobe size 64x64 pixels – 3 objects; strobe size 80x80 pixels – 1 object; strobe size 128x128 pixels – 1 object. With the above indicative values, the core has enough time to calculate the position of tracking objects in screen coordinates by the end of the next frame of the video sequence. If the core does not calculate the position of tracked objects by the end of the next frame, then to continue the calculation after frame change the core reads necessary segments of the new frame. This may occur when large strobe sizes are set and tracking is carried out of several objects simultaneously. If the performance speed of the selected FPGA type allows faster processing, then, the recommended sizes of tracking strobes and the number of objects tracked simultaneously will be larger. The above values are also limited by the speed of information exchange with DDR memory or BlockRAM and by specific features of the FPGA project. To ensure correct proper functioning of the device, the designer needs to assess these indicative values for a specific project.
Maximum and minimum sizes of tracking strobes	The maximum size of the tracking strobe is 128x128 pixels. The minimum size of tracking strobe is 16x16 pixels. The designer should provide for control of the input values of the tracking strobe sizes (the core should not be addressed with commands containing strobe sizes outside the range specified). The strobe size can be set separately for each object, both before object capture for tracking and during tracking. If you resize the strobe during the object tracking, the algorithm will automatically adjust its settings for each object independently. Thus, it is possible to resize tracking strobe when the size of object and its configuration change (when approaching, retreating or turning away) without rejection and re-capture of the object for tracking.
Maximum size of tracking object	Maximum size of the tracking object is limited by maximum size of the tracking strobe (128x128 pixels). If the object has a size larger than the maximum size of the tracking strobe, you can capture it by its part (e.g., by its most contrasting segment), and when the object moves away you can gradually adjust the strobe size to desired values.
Minimum size of tracking object	The core ensures stable tracking of objects of up to 8x8 pixels size. When tracking small-size objects it is recommended that objects occupy no more than a quarter of the strobe. Thus, when tracking the object with the size of 8x8 pixels the strobe size should be equal to the minimum value (16x16 pixels). When tracking large-area objects (80x80 pixels or more) it is recommended to choose the strobe size somewhat larger than the size of the objects itself.
Maximum movement	With a sudden change of the position of tracking object, the

Parameter	Values and notes
speed of tracking object	speed is no more than 20 pixels per frame in any direction. With a steady motion of the tracking object, the speed is limited by the capability of the tracker system used (drives speed). Such sudden change in the object position can occur in case of jittering of the optical axis of the video source. Thus, the core provides tracking of objects in case of frame jitter within the above limits. With a steady movement of the tracking object, the speed is limited by the capability of the tracker system used (drives speed). In the steady motion, the core algorithms perform calculation of velocity and acceleration of each object and, on this basis, organize their work. In this case, the speed of the object will be limited by the capabilities of the servo system (the speed of drives) and the characteristics of the video source (tracking break is possible because of blurred image of fast moving objects).
Minimum contrast of tracking objects	The core provides a stable object tracking with the contrast from 10%. This feature will greatly depend on the parameters of the background against which the object is observed. Stable tracking of objects against a uniform background can be achieved at a substantially lower minimum contrast than the value given above.
Maximum size of frames	The core operates with frames of up to 2048x2048 pixels size. The frame size is limited by the bus data width and available resources of the block memory (Block RAM).
Elongation of tracking objects	Tracking objects can have any configuration within the maximum size range of tracking strobes. The algorithms implemented in the core do not require any certain configuration of the tracking objects. Hence, tracking of any type of objects with any configuration is possible.
Output coordinates accuracy	Output coordinates of tracking objects are obtained with 1 pixel accuracy. The output information points to the position of the tracking objects in the screen coordinate system (relative to top left corner of the image).
Time lag of output coordinates	At the frame refresh rate of 25 Hz and recommended sizes of the tracking strobe, the core calculates the object coordinates before the end of the next frame in the video sequence (delay is 1 frame). For other FPGA types, a higher processing speed is possible, and there is a possibility of using video sources with a higher frame-repetition rate.
Format of processed images	The core works with images having color palette of 256 colors (grayscale) – 8 bits/pixel. Such format ensures optimum throughput when solving observation problem for distant objects and for observation of objects under poor light conditions where the use of color characteristics is not effective. Also, this format allows a considerably faster speed of reading images from the memory.
Максимальная частота кадров видео	Ядро обеспечивает корректную работу при частоте кадров до 30 Гц.
Maximum frame rate	Until for 30 Hz.

Note: the data above is given for FPGA Xilinx Kintes-7.

REQUIRED RESOURCES

Table 4 shows the resources required for the T-COR-11 IP core for various FPGA's of Xilinx company.

Table 4 – Resources required for the T-COR-11 IP core.

FPGA chip	Slice Rigisters	Slice LUTs	BRAM	DSP 48e1s
Kintex-7 xc7k325t-2ffg676	22103/407600 (5%)	17887/203800 (8%)	229/445 (51%)	80/840 (9%)
Artix-7 xc7a200t-2fbg676	21638/269200 (8%)	17840/134600 (13%)	229/365 (62%)	80/740 (10%)
Virtex-7 xc7vx330t-2ffg1157	2068/408000 (<1%)	2834/204000 (1.3%)	112/750 (14%)	

REQUIREMENTS FOR FGPA PROJECT

The use of T-COR-11 IP core does not impose any substantial requirements on the structure of FPGA project. The designer needs to provide the connection between the core and the buffer controller to store the video frame, register configuration, timing, control and reception of data from the core. The core reads video data from the controller via unified interface. The designer should ensure the transmission of video data according to the interface description. As an example, RIFTEK provides the source code of the frame buffer controller with the description. Figure 1 shows, an example of FPGA project with the use of IP core T-COR-11.

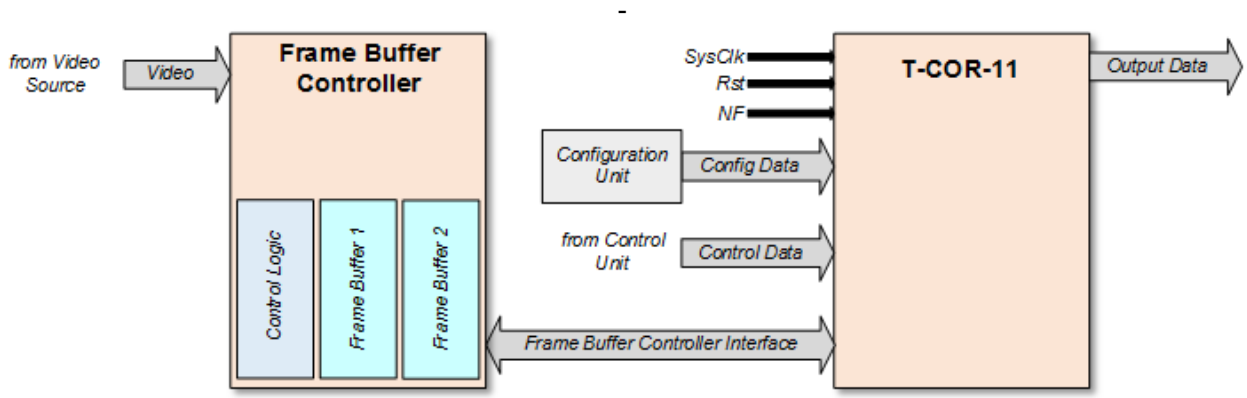

Figure 1 – Example of FPGA project with the use of IP core T-COR-11

The figure shows interaction between IP Core and FPGA project. The core is connected to the frame buffer controller by a unified interface. Frame buffer controller contains 2 frame buffers into block memory (Block RAM) and switches them when the next frame arrives. Frame buffer controller can store frames into DDR memory. The designer should ensure the working logic with DDR memory controller. Being switched on the T-COR-11 should be configured by writing configuration registers. Register configuration is carried out before work start with the use of special module (Configuration Unit). This module is developed independently and transmit data sequence necessary to configure the core. The designer can use a ready-made module supplied by RIFTEK with the core or develop it independently. Data recorded into configuration registers allows setting corresponding parameters before the work start. The designer should also foresee synchronization, core reset signal and new frame (NF) signal. NF signal indicates the current image is recorded and processing can be started. The designer ensures control on the core working process by sending control commands (Control Data) via the control interface. Information on capturing or removing objects from tracking or changing parameters of tracking, etc. is transmitted via this interface from the control module. Output data comes to consumer via unified interface (Output Data). This information can be transmitted to an executive subsystem, display subsystem, directly to the communication channel, etc. Data exchange interface with the T-COR-11 IP Core is described in the next section.

The developer should ensure one-way data transfer from the frame buffer controller to the core by the command from the core. It is also necessary to ensure information renewal in the frame buffer. Frame buffer controller can contain one or several frame buffers and logic to control them.

The controller supplied contains 2 frame buffers that are block memory IP cores. Control logic arrange sequential record of the frames into the buffers and data output on request from reading channels from the buffer which is not recording. Input interface is represented by clock frequency, controller ready flag to record data, data bus, data mask, data strobe, address bus (point the cell from which the record is made), record command strobe. Input data should arrive as lines of images on the data bus 4-byte width corresponding to the brightness of 4 consecutive pixels. Data on the data bus should be accompanied by continuous data strobe. The length of the data strobe in clock cycles must be matched by ¼ of the line length in pixels.

DESCRIPTION OF INTERFACE

Figure 2 shows the interface for interaction with the T-COR-11 IP core.

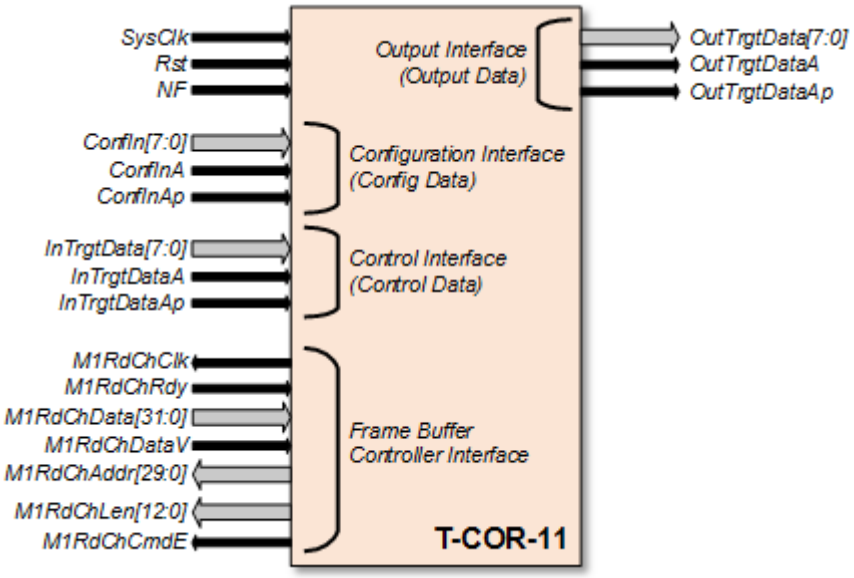

Figure 2 – Interface for interaction with the T-COR-11 IP Core

All input and output assignments are given in Table 5.

Table 5 – Input and output assignments.

Signal	Destination and notes
SysClk	Clock signal for the core. All signals for the core (control, configuration, configuration, reset, etc.) must be synchronized with SysClk. When testing for the FPGA Kintex-7 based project, SysClk frequency was chosen to be 200 MHz. The optimal clock signal frequency should be determined by the designer for a specific project.
Rst	Core reset signal. The signal must be synchronous with SysClk and have a duration of 1 clock cycle. The reset signal is used 1 time at the beginning of work of the core to write the configuration registers.
ConfIn[7:0] *ConfInA* *ConfInAp*	Interface for writing configuration registers. By this interface, the configuration registers are written before the work is started. Values of the configuration registers are passed via the ConfIn [7:0] bus. The ConfInA signal warns about data activity (data writing). ConfInAp signals the use of data in the registers. The interface is a unified one. Timing diagrams of the information exchange are given in Figure 3.
NF	New frame signal of 1 clock cycle duration. The signal must always come when video frame changes. Upon this signal, the core starts processing

Signal	Destination and notes
	of the next frame.
InTrgtData[7:0] *InTrgtDataA* *InTrgtDataAp*	Interface for control of the core. Through this interface commands are transmitted for the capture (rejection) of object tracking as well as for setting of the tracking parameters. The interface is a unified one and identical to the Interface for writing configuration registers. Timing diagrams are shown in Figure 3.
OutTrgtData[7:0] *OutTrgtDataA* *OutTrgtDataAp*	Interface for the output of information about the coordinates of tracking objects (Data output interface). This interface transmits the coordinates of tracking objects. The interface is unified with the control interface and configuration-register-writing interface. Timing diagrams of the information exchange are presented in Figure 3.
M1RdChClk	Clock signal for the frame buffer reading interface (all interface signals are synchronized or should be synchronous to the clock signal).
M1RdChRdy	Ready signal from frame buffer controller to receive reading requests.
M1RdChData[31:0]	Data read from the frame buffer.
M1RdChDataV	Read data validity strobe строб (is set when the data on the bus M1RdChData is active).
M1RdChAddr[29:0]	Address required data start.
M1RdChLen[12:0]	Length of required data in bytes.
M1RdChCmdE	Reading command strobe (one clock cycle M1RdChClk).

All the data exchange interfaces are synchronized to *SysClk* signal. Figure 3 shows the timing diagrams for the exchange of data via the data-output, control and register writing interfaces. The interfaces are identical in terms of information exchange principle used.

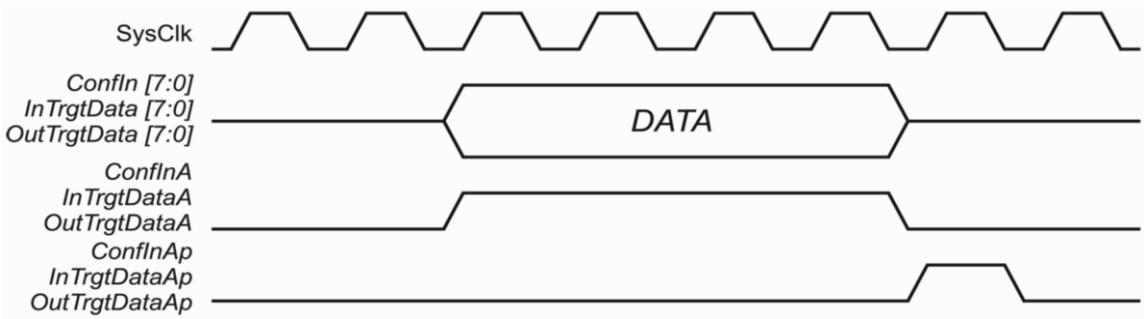

Figure 3 – Timing diagrams for the exchange of information via the coordinate data output, control and configuration register writing interfaces

Figure 4 shows the timing diagrams for the exchange of data with the frame buffer.

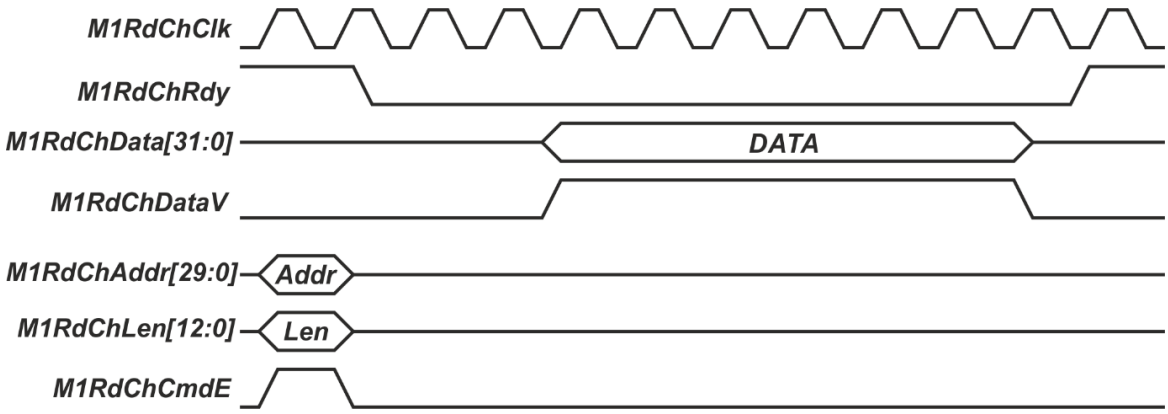

Figure 4 – Timing diagrams for the exchange of data with the frame buffer

PRINCIPLE OF INTERACTION WITH FRAME BUFFER

The designer should ensure one-way transfer of video data from the frame buffer to the core on request from the core. It is also needed to ensure the data in the frame buffer is being updated.

RIFTEK provides an example of the frame buffer based on the block memory (Block RAM).

Provided controller contains 2 frame buffers that are IP cores of the block memory. Control logic arrange sequential record of the frames into the buffers and data output on request from reading channels from the buffer which is not recording. Input interface is represented by clock frequency, controller ready flag to record data, data bus, data mask, data strobe, address bus (point the cell from which the record is made), record command strobe. Input data should arrive as lines of images on the data bus 4-byte width corresponding to the brightness of 4 consecutive pixels. Data on the data bus should be accompanied by continuous data strobe. The length of the data strobe in clock cycles must be matched by ¼ of the line length in pixels. Figure 5 shows the interfaces of the provided frame buffer.

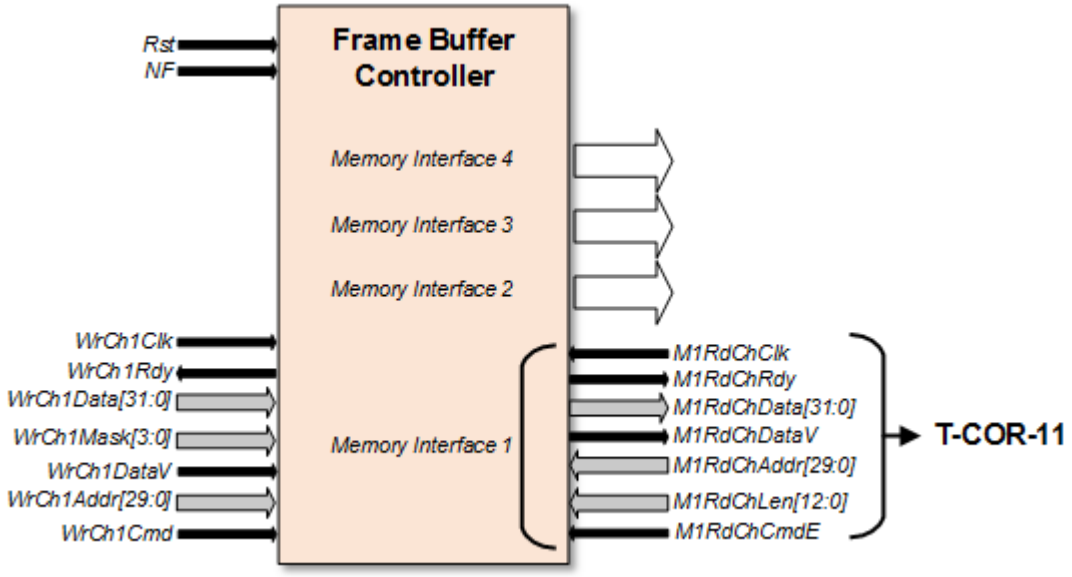

Figure 5 – Interfaces of the provided frame buffer

Assignment of recording in the frame buffer controller interface signals is shown in Table 6.

Table 6 – Interface signals of data recording in the frame buffer.

Signal	Assignment and notes
NF	Signal of a new 1 clock cycle length frame. Should arrive when a frame in a video sequence is replaced. At this signal the frame buffer controller changes the buffer for further recording. .
Rst	Frame buffer controller reset signal. the signal should be synchronized with SysClk and lasts for 1 clock cycle. The reset signal is applied 1 time at the beginning of core operation.
WrCh1Clk	Clock signal. All signals should be synchronized with it.
WrCh1Rdy	Ready-to-record signal.
WrCh1Data[31:0]	Data to be recorded. Four bytes of image data (8 bit per pixel).
WrCh1Mask[3:0]	Data mask for recording. For example, 0b0001 means 3 high-order bytes will be recorded.
WrCh1DataV	Data presence strobe in the line.
WrCh1Addr[29:0]	Data recording address.
WrCh1Cmd	Command to record the data directly in the buffers.

The module that controls image recording via the frame buffer controller must check ready-to-record signal *WrCh1Rdy* (Fig. 2). If the signal is set the module starts transferring of data and masks simultaneously setting data strobe (with the appearance of the data strobe ready-to-record signal will be off). If ready-to-record signal is not set, you should wait. After the data transfer was completed (1/4 line length in pixels), control module should reset data strobe and set the memory cell address (at which the record starts) into the address bus. After that should be given the command to start data recording in the frame buffer with *WrCh1CmdE* strobe of 1 clock cycle length of frequency *WrCh1Clk*.

The controller supplied has several (4) output interfaces to provide access from various modules of the project (video output via analog interface, video output via Ethernet, automatic tracking and recognition systems) to the frame buffers. Each output interface contains reading clock signal, controller ready-to-read flag, data bus, data strobe, address bus (pointing the cell at which the record starts), length of requested data bus, strobe of reading command. Address of reading may not be aligned to 4 bytes. Before forming read command the requesting module should check the readiness signal state *M'N'RdChRdy* (where N is the order number) and if it is set the module sets the address and length and give the reading command by *M'N'RdChCmdE* strobe of 1 clock cycle length of frequency *M'N'RdChClk* (readiness flag is reset). After preparation the controller installs the requested data on the data bus and forms the data strobe. After data transfer completion, the frame buffer controller forms end of reading strobe *M'N'RdChDone* of 1 clock cycle length of frequency *M'N'RdChClk*.

By default the frame buffer controller is configured for the image size 720x576 pxl 8 bit per pxl. format. To change the frame buffer size for the other projects it is needed to open the project tree in Xilinx ISE 14.7 developer environment, select BRAMFrameController1 and double-click to select FrameBRAM1. Dialog box to configure the block memory modules is open as shown in Figure 6.

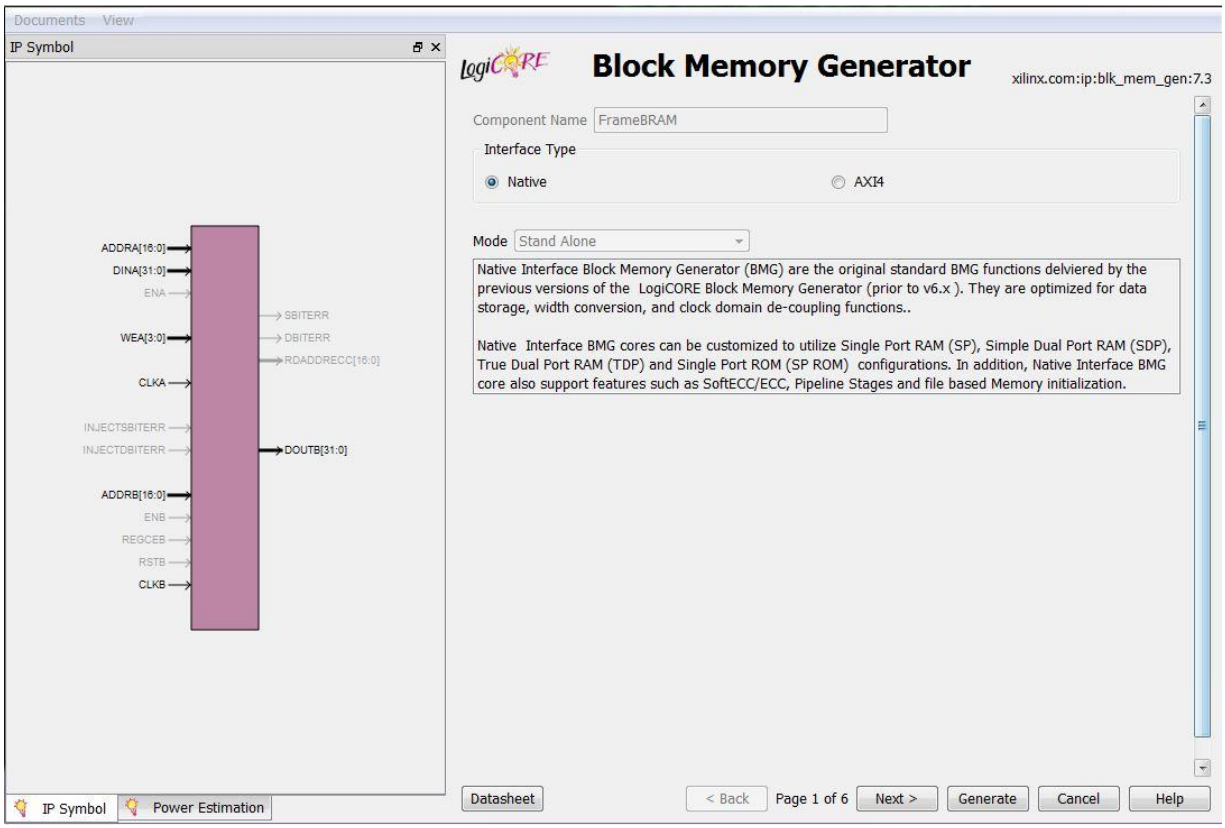

Figure 6 – Block memory module (BRAM) configuration dialog

It is necessary that the parameters have been set in the same way. Then click «Next». On the next page of the dialog box set the parameters as shown in Figure 7.

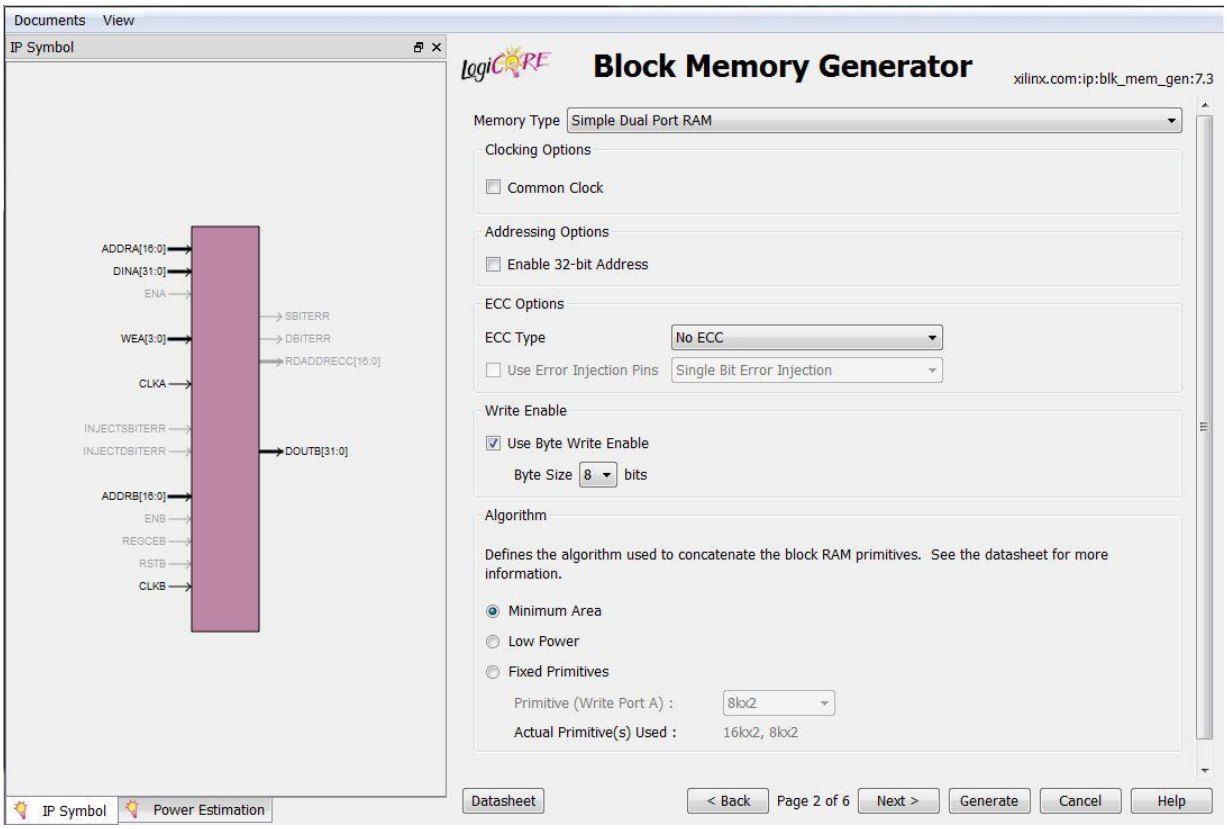

Figure 7 – Block memory module (BRAM) configuration dialog

Then click «Next». On the 3rd page of dialog box set the parameters shown in Figure 8. WriteDepth parameter should be set according to the size of the required frame buffer (he number of pixels in the frame in format 8 bit per pxl. divided by 4).

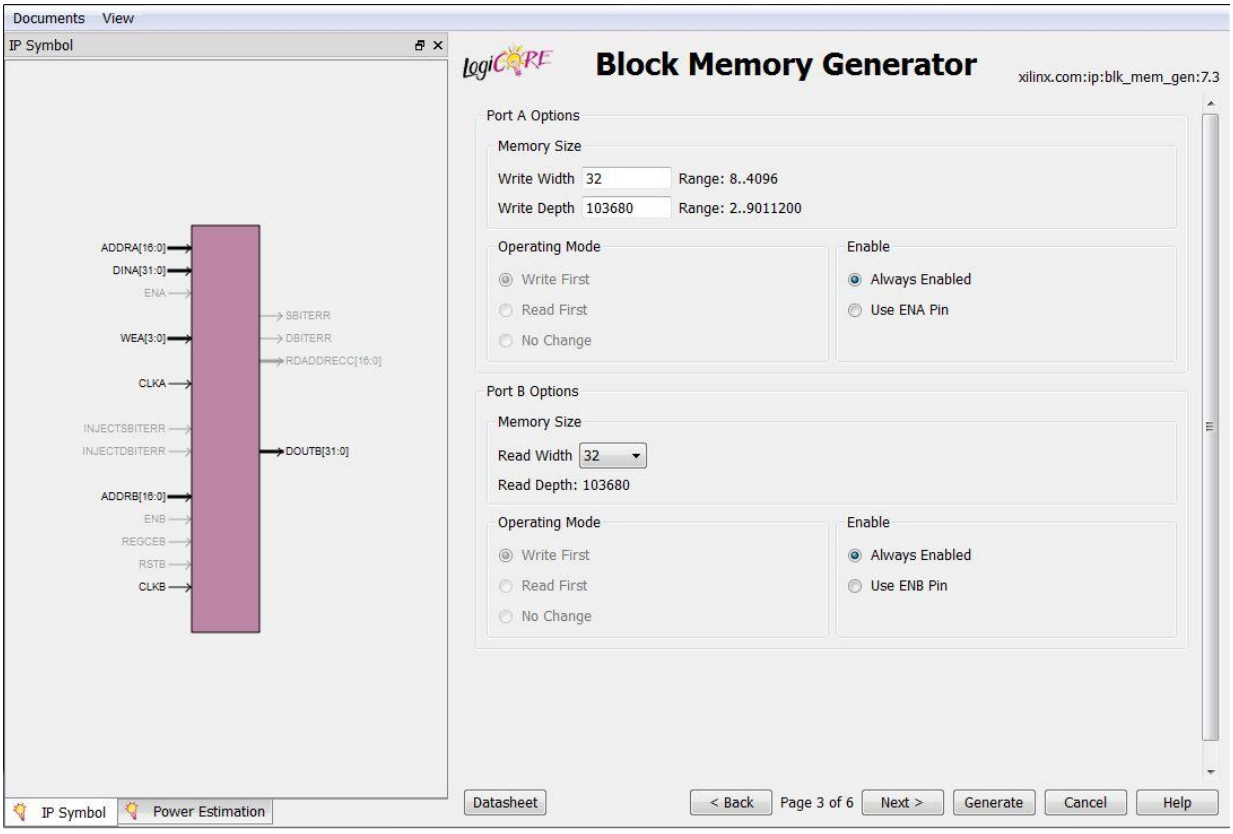

Figure 8 – Block memory module (BRAM) configuration dialog

CONFIGURING THE CORE

Before the work is started, the T-COR-11 IP core should be configured. Configuring consists in the writing of the configuration registers. The user is given an option to carry out parameter setting of the core algorithms by himself. Configuration is performed through the unified interface for writing configuration registers as described in the previous section. This interface works with the signals: ConfIn [7:0], ConfInA, ConfInAp. The IP core parameters available for setting are shown in Table 7.

Table 7 – Parameters of the IP core available for setting.

Parameter name	Designation	Value	Assignment
Frame buffer width	FBW	2 bytes (permitted values from 8 to 2047)	Frame buffer width in pixels (width of images arriving from video source).
Frame buffer height	FBH	2 bytes (permitted values from 8 to 2047)	Frame buffer height in pixels (height of images arriving from video source).
Search area width	CW	1 byte (permitted values from 4 to 64)	Full width of the region where search of tracking object will be performed relative to a certain extrapolated point.
Search area height	CH	1 byte (permitted values from 8 to 2047)	Full height of the region where search of tracking object will be performed relative to a certain extrapolated point.
Detection threshold	P	1 byte (permitted values from 0 to 255)	This parameter defines the detection threshold from 0 to 255 which is equivalent to a range from 0 to 1.
Parameter 1 of the object path smoothing	K1	1 byte (permitted values from 0 to 255)	This parameter defines the contribution of the current measured position of the tracked object to the calculation of smoothed values.
Parameter 2 of the object path smoothing	K2	1 byte (permitted values from 0 to 255)	This parameter defines the contribution of the speed of the tracked object to the calculation of smoothed values.
Parameter 3 of the object path smoothing	K3	1 byte (permitted values from 0 to 255)	This parameter defines the contribution of the acceleration of the tracked object to the calculation of smoothed values.

To configure the core, it is necessary to transmit 34 bytes of the configuration data (configuration packet) via the corresponding interface before the work is started (at the time of

power-up). After this, the core is ready for operation. If there is a need to change the parameters in the process of work (e.g., change smoothing factor or the width and height of the region under check), it is possible to re-write the parameters promptly during tracking without turning off or resetting the core. Table 8 shows the sequence of configuration data in the configuration packet.

Table 8 – The sequence of the configuration data

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Data	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Byte	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
Data	0	0	0	0	0	0	0	FBW		FBH		CW	CH	P	K1	K2	K3

Explanations with respect to each of the parameters are given below.

Frame buffer width. This width corresponds to the width of images under processing. The frame buffer width is measured in bytes, and with the format of images of 8 bits/pixel corresponds to their width. In addition, these values are used by the core to restrict the movement of tracking strobes within the image boundaries (the strobe cannot go beyond the image boundaries. This parameter can be changed by user in the process of work.

Frame buffer height has the same physical sense as the frame buffer width.

Search area width and height. These parameters are measured in pixels. These values define the dimensions of an image region relative to a certain point (unique for each object) where search of tracking objects is carried out in the next frame. These values affect the core operation speed. With higher values, the number of search positions in the image strongly increases which requires additional time for processing. However, the algorithms work well under conditions of image jitter. With small values of the search area width and height, the core will work at highest speed but in this case frame jitters will go beyond the limits defined by these parameters which may result in tracking rejection. It is recommended to use the following value settings:

for 1 tracking object with a size of 128x128 pixels – the maximum search area is 23x23 pixels;

for 1 tracking object with a size of 80x80 pixels – the maximum search area is 33x33 pixels;

for 2 tracking objects with sizes of 64x64 pixels – the maximum search area is 23x23 pixels;

for 2 tracking objects with sizes of 40x40 pixels – the maximum search area is 33x33 pixels;

etc.

The recommended values can vary considerably depending on FPGA type used. Our data are presented for FPGA Kintex-7. The parameters can be changed by user during tracking depending on the number and size of tracking objects.

Detection threshold. This parameter changes in the range from 0 to 255 which is equivalent to the threshold probability of object detection in the image from 0 to 1.

If it turns out that in the course of tracking the probabilities of finding the object in all search positions (search area) are lower than the threshold value, the core switches to the path prolongation mode (extension of the tracking object path based on the calculated parameters of its movement) until the probability of finding object in a given point is above the threshold (automatic re-capture of the object for tracking).

Object path smoothing parameter. The parameters of smoothing of the tracking object paths define the contribution of the object movement parameters (current position measured, speed and acceleration) to the calculation of the object position. The parameters are set in the range from 0 to 255 which is equivalent to the values from 0 to 1. For intensively maneuvering objects, it is recommended to use high values of K3 and for steadily moving objects high values of K2 are recommended. The parameters should be set depending on the conditions of work. It is advised to use the following mean values: K1 – 179, K2 – 125, K3 – 61. Like other parameters, these can change in the process of work.

If it is necessary to change the core parameters during work, the whole configuration packet should be transmitted together with all changes required. In order to configure the core more conveniently, it is recommended to synthesize a special module (core) which will transmit the required messages to the T-COR-11 IP core. If the core parameters need to be changed quickly and efficiently, this function should be assigned to the control module of the whole FPGA project.

CONTROL OF OPERATION MODES

Control of the operation modes of the core is carried out via a unified control interface (signals *InTrgtData[7:0]*, *InTrgtDataA*, *InTrgtDataAp*). The T-COR-11 IP core performs capture of objects for tracking by an external command. The command may contain the instruction for capture of an object for tracking or for rejection of tracking. Each object (up to 5 objects in all) has its own commands that arrive. The command format is given in Table 9.

Table 9 – Format of a command to control the operation mode.

Byte	0	1	2	3	4	5	6	7	8	9
Data	NC	Cmd	X		Y		W	H	DX	DY

Table 10 presents types of data contained in the command and their assignments.

Table 10 – data types and their assignments.

Byte №	Designation	Default value (HEX)	Description
1	CnannelNum	0x00	Number of tracking channel from 0x00 to 0x04 (In total 5 channel supported). Parameters of the tracking channel to be modified should be specified in the field.
2	Command	0x00	Code of the command (action code) for the tracking channel. The following commands are valid: 0x00 – adjust the position of the tracking strobe (is processed in MO mode only), 0x01 – capture for tracking, 0x02 – remove from tracking, 0x03 – switch to extension path mode, 0x04 – switch off extension path mode (automatic recapture, if the object is find in the search strobe), 0x05 – adjust the tracking strobe position (the position of the strobe is adjusted in AT mode in increments of 2 2 pixels in a specified side), 0x06 – adjust the tracking strobe size.
3	X	0x00	High byte of the horizontal coordinate of the tracking strobe center.
4	X	0x00	Low byte of the horizontal coordinate of the tracking strobe center.
5	Y	0x00	High byte of the vertical coordinate of the tracking strobe center.
6	Y	0x00	Low byte of the vertical coordinate of the tracking strobe center.
7	W	0x10	Width of the tracking strobe of the given channel.
8	H	0x10	Height of the tracking strobe of the given channel.
9	DeltaX	0x00	Horizontal offset to adjust the tracking strobe position in AT mode. Values can be either positive or negative (1 in the high bit indicates a negative value). Offset is allowed no more than 4 pixels for one command. Increment allowed – 0x02.
10	DeltaY	0x00	Vertical offset to adjust the tracking strobe position in automatic tracking mode.

Note: All coordinates should be given according to window coordinate system (0;0) – left upper corner of an image.

Thus, by using a single command it is possible to control tracking and change parameters of the process. The control sequence should be as follows. If you want to capture an object for

tracking, select the channel number and send a capture-for-tracking command. If you want to change the tracking parameters for a given channel, you should send an appropriate command.

OUTPUT INFORMATION

The output information from the IP core T-COR-11 is available via a unified interface for receiving data from the core (*OutTrgtData[7:0], OutTrgtDataA, OutTrgtDataAp*). The data exchange principle is analogous to that used in the configuration-register-writing interface and control interface, the only difference being in the data transmission direction. The output information represents packets 11 bytes long. The information is transmitted for each video frame (1 data package per video frame). Structure of output data packets is presented in Table 11.

Table 11 – Structure of output data.

Byte №	Designation	Default value (HEX)	Description
1	StatusFlag	0x00	Channel state flag: 0x01 – the channel is busy (AT mode), 0x00 – the channel is free (AT mode is off).
2	ProlongationFlag	0x00	Extended path mode flag 0x00 – path extension is off, 0x01 – path extension is on. Extended path flag is actual in IT mode only. If the channel is free extension flag should be set to 0x00.
3	ProlongationReverseCount	0x00	The countdown from the start of trajectory extension. At the initial moment value is set to 0xFF when the value 0x00 is reached the tracking is forced to reset
4	X	0x00	High byte of the horizontal coordinate of the tracking strobe center.
5	X	0x00	Low byte of the horizontal coordinate of the tracking strobe center.
6	Y	0x00	High byte of the vertical coordinate of the tracking strobe center.
7	Y	0x00	Low byte of the vertical coordinate of the tracking strobe center.
8	StrobW	0x00	Width of the tracking strobe of the given channel.
9	StrobH	0x00	Height of the tracking strobe of the given channel.
10	VelX	0x00	High byte of horizontal smooth component of the tracking strobe displacement speed in AT mode.
11	VelX	0x00	Low byte of horizontal smooth component of the tracking strobe displacement speed in AT mode. To get the displacement speed in «pixel/frame» grade divide the obtained value by 256.
12	VelY	0x00	High byte of vertical smooth component of the tracking strobe displacement speed in AT mode. To get the displacement speed in «pixel/frame» grade divide the obtained value by 256.
13	VelY	0x00	Low byte of vertical smooth component of the tracking strobe displacement speed in AT mode.

Byte №	Designation	Default value (HEX)	Description
14	TargetSubStrobX	0x00	Horizontal coordinate of the upper left corner of the target strobe within the tracking strobe. Coordinate system begins in the left upper corner of the tracking strobe. **IMPORTANT!** For the current version of the protocol, the value is always equal to 0x00.
15	TargetSubStrobY	0x00	Vertical coordinate of the upper left corner of the target strobe within the tracking strobe. Coordinate system begins in the left upper corner of the tracking strobe. **IMPORTANT!** For the current version of the protocol, the value is always equal to 0x00
16	TargetStrobW	0x00	Width of the target strobe within the tracking strobe. The parameter is used to separate the object from the background within the tracking strobe for the further maintenance or other adjustments. **IMPORTANT!** For the current version of the protocol, the value is always equal to 0x00.
17	TargetStrobH	0x00	Height of the target strobe within the tracking strobe. The parameter is used to separate the object from the background within the tracking strobe for the further maintenance or other adjustments. **IMPORTANT!** For the current version of the protocol, the value is always equal to 0x00.
…	…	…	(Repeating of another 4 blocks of 17 bytes for the next 4 tracking channels)

CONNECTING THE CORE TO FPGA PROJECT

Since the core is supplied for the FPGA Xilinx project, the procedure of connection of the T-COR-11 IP core is presented for IDE ISE 14.7. The core represents a file with extension *.ngc and a file with extension *.v (On request, we can supply *.Vhdl) (TrackingProcessor.ngc, TrackingProcessor.sym, TrackingProcessor.v (.Vhdl)). The file TrackingProcessor.ngc is the synthesized core. The file TrackingProcessor.v (.Vhdl) is the shell where the core inputs and outputs are specified. Below is the procedure of using the core in an FPGA project with a HDL upper level in IDE Xilinx ISE 14.7.

To use the core in the projects where the upper module is in the form of an HDL-file, follow these steps:

1. Add the file TrackingProcessor.v (.vhdl) to the FPGA project.
2. Mark the file added and run «View HDL Instantiation Template» in the section ‘Processes’.
3. Copy the core template from the open window to the upper module.
4. Connect the inputs and outputs of the core with the inputs and outputs of the frame buffer controller core and control core (the control interface, configuration-register-writing interface and data-receive interface are connected with the interfaces of the (core) control module).
5. In the project synthesis parameter «Cores Search Directories», specify the path to the folder containing the file TrackingProcessor.ngc.

To simplify the integration of the core into finished projects, we supply project files together with Verilog file of the core configuration module TPConfig.v, which is linked with the definition file 'Definitions.v.' The configuration module is connected with the configuration-register-writing interface and makes it possible to configure the core with appropriate parameters before the work is started. The designer can set the required parameters in the file before he starts to synthesize the project. It is recommended to organize the initial configuration using a single control module which allows changing of settings in the process of work. The configuration module can be directly connected to the core 'TrackingProcessor" via the configuration-register-writing interface. The core is supplied together with the frame buffer control module BRAMFrameController.v and the file Xilinx CoreGen FrameBRAM.xco.

CONTACTS

RIFTEK – a private company dealing with development and production of optoelectronic devices intended for measuring geometric quantities and video processing tools.

Logoiski Trakt 22-311, 220090 Minsk, Republic of Belarus

Tel./fax: +375 17 281-35-13; +375 17 281-36-57
GSM: +375 29 655-72-55
e-mail: info@riftek.com; sales@riftek.com
Website: www.riftek.com