What is NXTCam

NXTCam is a real-time image processing engine. Think of it as a vision sub-system with on-board processor and a protocol interface that is accessible through a standard NXT sensor port. This interface provides high-level, post-processed information of the image NXTCam sees. The processed information contains the bounding box coordinates of the objects of interest in view of NXTCam. In line tracking mode, this information contains coordinates of line segments.

NXTCam does not send the image itself to NXT. However, connecting the NXTCam to a PC via USB and NXTCam View, you can see the image on the PC.

NXTCam-v4 is compatible with NXTCam-v3 software and libraries.

NXTCam Specs

- Track up to 8 different colorful objects at 30 frames/second
- Configure the NXTCam using USB interface on Windows XP, Vista, 7, or 8.
- Supports two tracking modes: Object tracking and Line tracking.
- Provide real-time tracked object statistics (number of objects, color of objects, bounding box coordinates or line coordinates) through a standard NXT sensor port.
- Tracked image resolution of 88 x 144 pixels at 30 frames/second
- Perform full-resolution (176 x 144) pixels color image dumps to PC via USB port.
- Maximum power consumption (42 mA at 4.7 V)
- Uses NXT compatible I2C protocol for communications.
- Supports Auto Detecting Parallel Architecture (ADPA) for NXT sensor bus. This means that NXTCam can coexist with LEGO or third party digital sensor on the same NXT port. ADPA support enables user to employ several sensors on the same port without the need of external sensor multiplexer, reducing the overall size without compromising the functionality.

What you will need before using NXTCam

Connector Cables

For runtime operations (on the robot, in autonomous mode), the NXTCam connects to NXT on a sensor port using a standard connector cable that comes with NXT (the one with jacks similar to phone jacks).

For offline operations (for programming and configurations), NXTCam connects to PC using mini-USB cable. (This is in addition to the cable you would use to connect NXT to your PC). Adjacent picture shows the mini-USB connector you would need on your USB cable, this connector is commonly used for digital cameras. If you need to acquire a cable separately, it should be a '5 wire' cable.

USB Driver installation

In order for NXTCam to work properly, you will need to install USB drivers for your operating system. Currently support exists for:

- Windows XP (i386 and AMD processors)
- Windows Vista (i386 and AMD processors)
- Windows 7 (i386 and AMD processors)
- Windows 8 (i386 and AMD processors)
- Mac OS X v10.4.10 (PowerPC G4)

Download the drivers and installation instructions from following location: http://www.mindsensors.com/content/20-nxtcam-usb-driver-installation-instructions

Viewer and Configuration Software

To see the picture that's in the field of view of NXTCam, capture that picture for analysis and configure the Colormaps for onboard processing, you will need to install and use Viewer and Configuration software on your PC.

Download Viewer and Configuration software for your operating system from following location:

For MS-Windows XP/Vista/7/8: http://nxtcamview.sourceforge.net/ (developed by Paul Tingey). Note that, on newer computers it may require to install .NET version 2.0.50727. It's Ok to install that version.

For Mac OSX 10.5 or older:

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=13

(This Mac software does not work with newer OSX versions).

For Linux:

USB Configuration on Linux:

The NXTcam used FTDI chip for USB communication. This chip implements USB serial interface and uses FTDI serial drivers (these are commonly available in any Linux).

To configure USB, NXTCam uses FTDI chip with manufacturer ID as 0403,and Device ID as ABB8

In your Linux installation, you will have to configure your USB enumerations based on these values.

Viewer software on Linux:

The NXTCam is compatible with AVRCamView software and that uses Java. It has been tested on Redhat Linux and should work just as well on other Linux variants.

download source at:

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=13

(derived from https://github.com/jgraef/aNXTCam developed by Janosch Gräf)

Note: If the Java program is unable to find the USB device, you may have to specify the USB device in the code.

Programming Environment(s)

EV3:

To use capabilities of the sensor, please download EV3 blocks available at following URL:

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=137

Installation instructions for EV3 block are available at:

http://www.mindsensors.com/content/13-how-to-install-blocks-in-ev3

Download EV3 sample program from following URL and modify it to suit your needs.

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=138

Note: While using with EV3, ensure to use firmware version 1.03H or 1.03E or higher on your EV3.

NXT-G:

NXTCam is supported for use in NXT-G using a custom block. Download the block from following URL:
http://www.mindsensors.com/index.php?controller=attachment&id_attachment=23

This block provides functionality for tracking objects based on the Colormaps downloaded on NXTCam.

Installation instructions for NXT block are available at:

http://www.mindsensors.com/content/21-nxt-g-blocks-how-to-install-blocks

Note: While using with NXT-G, ensure to use latest firmware version on your NXT (1.31 as of this writing).

LeJOS API's are available at:

http://lejos.sourceforge.net/p_technologies/nxt/nxj/api/lejos/nxt/NXTCam.html

RobotC API's are available at:

https://github.com/botbench/robotcdriversuite

NXC/NBC Library functions are available at:

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=94

Connecting NXTCam

Wiring for NXTCam

NXTCam may be connected to any of the sensor ports of NXT using standard NXT connector cable. In your program, select the appropriate port number to which NXTCam is connected.

⚠ **WARNING**

Do not connect the NXTCam to any motor port, as the voltage applied by the motor port may damage the electronics of NXTCam.

During offline operations, such as programming and configuration, NXTCam must be connected to PC (using USB cable) as well as NXT (using standard NXT connector cable) while NXT is powered ON.

During runtime (or autonomous) operations on NXT, the USB connection to PC must be removed.

NOTE: While NXTCam is connected to NXT as well as PC, the PC communication takes priority over any other communication.

In other words, while NXTCam is connected to PC as well as NXT, if you run a program on NXT, it will not be able to talk to NXTCam.

Mounting NXTCam on your contraption

The holes on the NXTCam enclosure are designed for tight fit of Technic pins (or axles) with '+' cross section. The holes however are not designed for repeated insertions/removals of these pins.

To mount NXTCam on your contraption we suggest that you use two dark gray 'Technic Axle 3 with Stud' as shown.

Insert axles from the front (lens side) of the NXTCam and secure with a bushing on the back or mount it on your contraption directly.

Alternately, you may use blue 'Technic Axle Pin with Friction', as shown.

While disassembling contraption, leave the axles and/or pins on NXTCam.

LED on NXTCam

The LED on NXTCam blinks initially when NXTCam is booting up (when power is connected through USB port or NXT Sensor port). During the normal operation the LED may blink occasionally.

Configuring NXTCam Colormaps using NXTCamView

What's a Colormap?

The objects of interest are recognized by NXTCam by matching the stored color values with the captured image. To do that, color values of the objects of interest need to be stored in NXTCam's memory. These color values are known as Colormaps.

NXTCam can store up to 8 Colormaps and provide processed information of the objects matching those Colormaps.

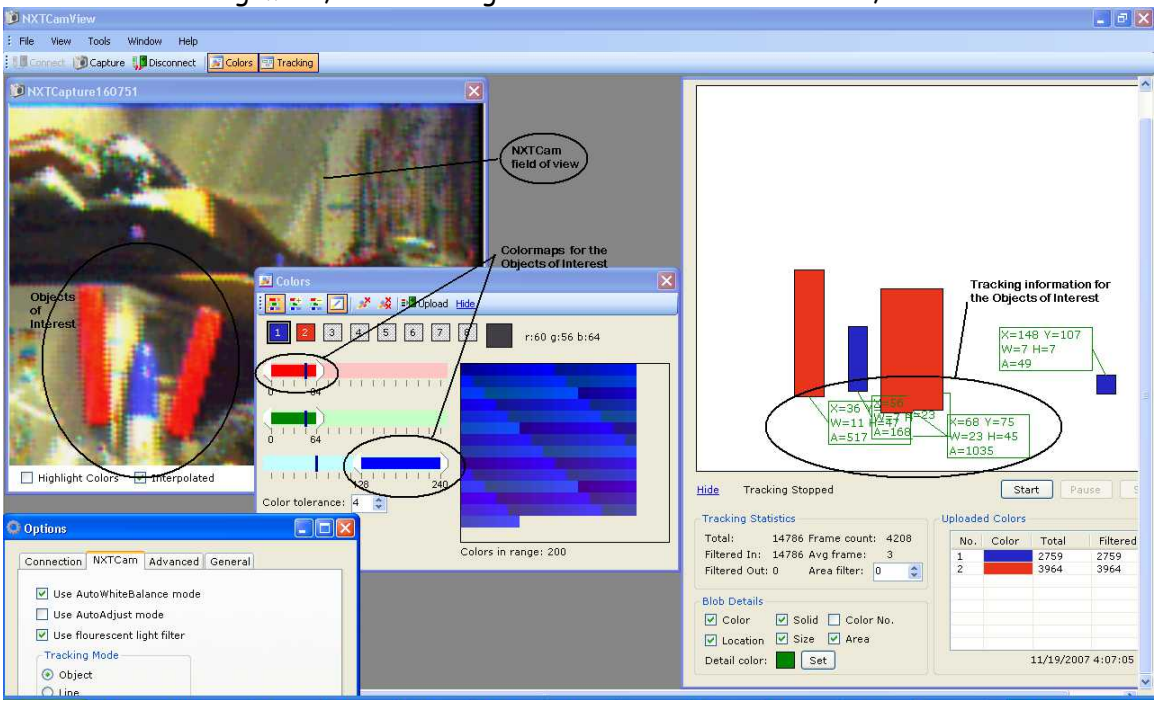
Default Object Colormap

The NXTCam is shipped with a default Colormap to track a light source (white light). Download a test program or write a test program, and try tracking a light source.

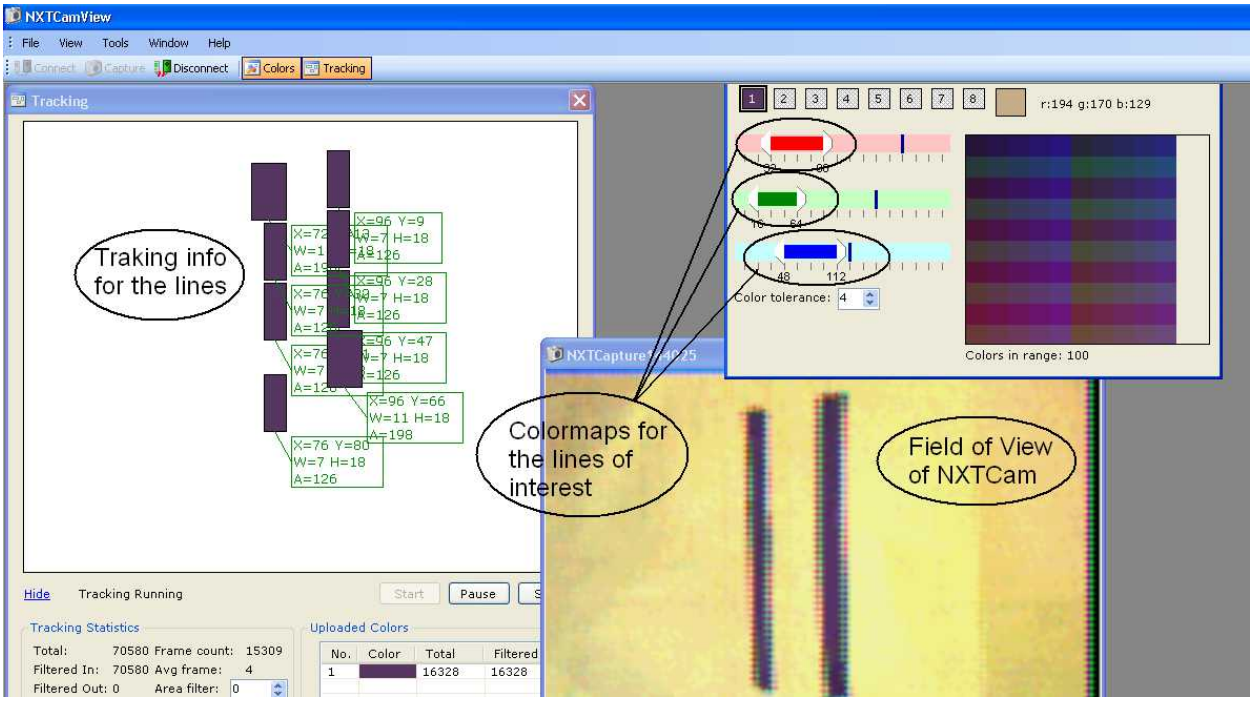
Objects of interest

Below are screen dumps of NXTCamView software, showing objects of interests and their tracking information.

The top left window in the picture below shows the field of view of NXTCam. The objects of interest from this view are the red and blue pens. In the object tracking mode, the bounding box coordinates are returned, as shown:



In line tracking mode, beginning point and end point coordinates of a line representing the object are returned. In the following picture, the bounding boxes are drawn for the line coordinates received from NXTcam:



To pick Colormap values of your objects of interest and to store them on NXTCam using the NXTCamView software, please follow these steps:

1. Download, install and start NXTCamView on your PC.
2. Ensure the two NXTCam drivers (USB-to-Serial and NXTCam) have been installed on your PC.
3. Ensure the NXTCam is plugged into one of your PCs USB ports.
4. In NXTCamView, setup communications to the NXTCam via **Tools\Options**. Often only the COMPort needs to be set.
5. **Connect** to your NXTCam.
6. **Capture** an image (or two) to check the focus and lighting conditions.
7. **Select** some colors to track by clicking on the image you captured. Colors are shown as ranges of red, green and blue (min/max values of each). You can add or remove colors from a range by holding down the "CTRL" key or "SHIFT-CTRL" keys while clicking.
8. **Upload** the color ranges into the NXTCam.
9. Test your colors in the **Tracking** window to see what object blobs are returned.
10. **Disconnect** NXTCamView from your NXTCam.

Also watch a demo video 'Capture and Select' at following URL:
http://nxtcamview.sourceforge.net/DemoScreenCam.htm

Tips on using NXTCam in your environment

Object Colors in Line Tracking Vs Object Tracking Modes

In the object tracking mode, you can track objects comprising of 8 distinct colors. While selecting colors, avoid any overlap between colors of different objects.

In line tracking mode, only the first color from the Colormap is used, and it is recommended to limit the number of colors to one.

Lighting conditions

The NXTCam is designed to operate under **white fluorescent light**. If you notice reddish image color, which tends to happen when your environment has lot of ambient Infrared light, try to find the source of Infrared light and reduce it by replacing it with fluorescent light.

For advanced operations it is possible to change color gain, brightness and contrast of NXTCam by manipulating the I2C register values.

Focus

As a factory default, NXTCam lens is set for optimal focus between 2 and 4 feet. The lens is screwed in it's holder and it is designed to be tight to prevent accidental rotation and loss of focus. To refocus the lens, gently turn the lens from the holder, capture images and see if the new focus is satisfactory. Do not apply excessive force, as it may damage the lens. For better grip while turning lens, you may wrap a rubber-band around the exposed threads of the lens. To check the focus, you can use Viewer software and perform a 'capture' to see the new focus results.

Updating your Colormap

Human eye (and brain) is conditioned to adapt to ambient light conditions and see. Whereas based on ambient light, the colors of objects appear different to a camera CCD. In other words, a blue ball in your laboratory lighting conditions will appear to be a different shade of blue in Gymnasium lighting. Considering this aspect, ensure to update your NXTCam Colormap based on your final lighting conditions.

Selecting Colors of your Target Object for Colormap

Due to angle of light and shadows cast on the object, to NXTCam, the object appears to be a mix of dark and light shades. In choosing color for your colormap, try to select the lighter shade of your object's colors. If with lighter shades you have trouble tracking darker shades, then select complete range of colors.

Timeout

Once the tracking mode is started, your program should read the object information from NXTCam within 9 seconds and continue to read at least at that interval. If there is inactivity of 9 seconds (or more), the tracking mode is stopped.

Troubleshooting NXTCam communication

Troubleshooting NXTCam communication with your PC

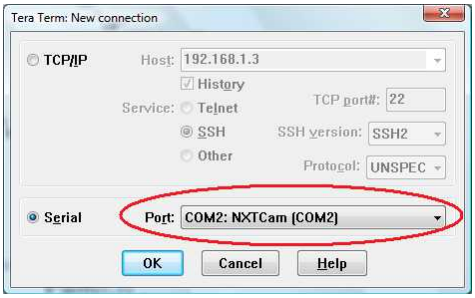
To ensure USB drivers are installed properly, follow these steps:

1. Ensure to install the USB drivers as mentioned in this document.
2. Connect the NXTcam using a USB cable to your computer.
3. From Start menu -> Right click on Computer, select 'Properties', select 'Hardware' tab, and select the 'Device Manager'.
4. Expand the 'Universal Serial Bus Controller' entry.
5. You should see 'NXTCam' listed.
6. In the same Device Manager, Expand 'Ports (COM & LPT) entry.
7. You should see a COM port listed for NXTCam (this is the COM port you should use for your viewer software configuration).

To ensure actual USB communication, follow these steps:

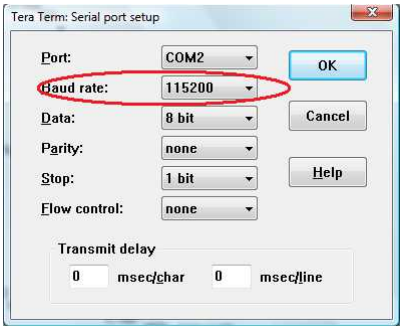
Install TeraTerm from following URL:

https://ttssh2.osdn.jp/index.html.en



Select NXTCam port from the list provided.

Setup Baud Rate as shown:



In the main terminal window, just press <enter>
You should get a response from NXTCam as 'NCK'

In the same window, type:
GV<enter>
You should get a response from NXTCam as 'NXTcam V n.n' (where n.n is your NXTcam version number)

Alternately, you can use Hyperterminal (on Windows XP)

1. Connect the NXTcam using a USB cable to your computer.

2. Run Hyperterminal: Start -> All Programs -> Accessories -> Communication -> Hyper Terminal
3. Give a name for the configuration, say 'mynxtcam'.
4. In the next dialog box specify:
   Connect Using: <NXTCam COM Port> (the port you noted from Ports (Com and LPT) entry in Device Manager)
5. Click OK.
6. In next window Specify:
   Bits per second: 115200
   Data bits: 8
   Parity: None
   Stop bits: 1
   Flow Control: None
7. Click OK.
8. In the main terminal window, just press <enter>
   You should get a response from NXTCam as 'NCK'
9. In the same window, type:
   GV<enter>
   You should get a response from NXTCam as 'NXTcam V n.n' (where n.n is your NXTcam version number)

Troubleshooting 'Time-out' error

While using NXTCam-v4 on PC (with NXTCamView), if you get 'Search Error: The operation has timed out', NXTCam-v4 AVR firmware may be corrupted.

As a feature of NXTCam-v4, the firmware is designed to be re-writable, but sometimes due to unknown reason it writes itself and corrupts.

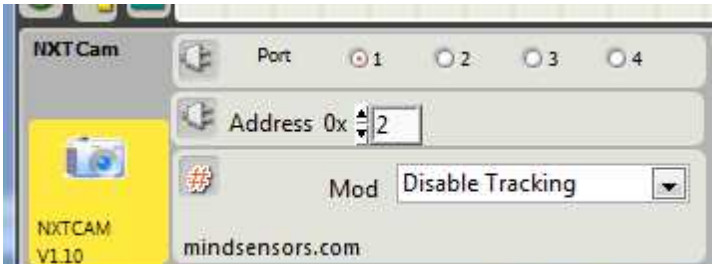
In this case, the firmware needs be re-written by following procedure in section 'How to change Firmware of AVR Mega 8 processor' of 'Advanced Programming Guide'.

Troubleshooting NXTCam communication with your NXT

Ensure to install NXT-G blocks for NXTCam as mentioned above in the document.

1. Connect your NXT to your PC using it's standard USB cable.
2. Connect your NXTCam to NXT Port 1 using standard NXT sensor cable.
3. Run NXT-G software on your computer.
4. Power on the NXT.
5. Start a New Program (say Untitled-1), and from Advanced Block Palette, drag and drop the NXTCam block in it.

6. Click on the block, and examine the bottom left corner of NXT-G window (where block control panel is located).
7. You should see NXTCam version number shown in the bottom left corner, as shown in the adjacent picture.



8. If the NXTCam is not connected correctly to your NXT, this status will indicate 'No Device'. If that happens, ensure the port and Address in your program match to what's on NXTCam.

Advanced Reference Information:

Appendix A: I2C

Open Source Software and Hardware

NXTCam is based on AVRcam, and is Open Source using GNU license. We encourage you to improve the source code and features and inform us the changes for inclusion in future releases. DownloadAdvanced Programming Resources and documentation from the following url:

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=326

I2C Operations

Pins used: SDA(1), GND(2), SCL(3), +5V(4)

Following table lists the register definitions and setup commands:

Commands		Action
ASCII	**Hex**	
A	0x41	Sort tracked objects by size
B	0x42	Select Object tracking mode
C	0x43	Write to image sensor Registers Use CAUTION when using command C since this can stop NXTCam from working properly. In case this happens, please power off your NXTCam and power it on again.
D	0x44	Disable Tracking
E	0x45	Enable Tracking
G	0x47	Get the Color map from NXTCam Engine
H	0x48	Read data from the image sensor Registers
I	0x49	Illumination on (Future)
L	0x4C	Select Line tracking mode
N	0x4E	Set ADPA mode **O**N (setting stored in NVRAM)
O	0x4F	Set ADPA mode **O**ff (default) (setting stored in NVRAM)
P	0x50	Ping NXTCam Engine
R	0x52	Reset NXTCam Engine
S	0x53	Send the color map to NXTCam Engine
T	0x54	Illumination Off
U	0x55	Sort tracked objects by color
V	0x56	Get NXTCam Engine firmware version No, Read resulting string at 0x42 (12 bytes).
X	0x58	Do not Sort tracked objects
J	0x4A	Lock tracking buffer Tracking is going on continuously, and while you read

		buffer may be updated by the tracking engine. You can issue this command to Lock the buffer updates. After issuing this command allow 25 milliseconds for any updates in progress to finish. Then read the tracking information.
K	0x4B	Unlock tracking buffer. If you had locked the buffer before reading, ensure to unlock it after you are done reading.

While programming in NXC, you can use the API function NXTCam_SendCommand () to send the command to NXTCam. While using NXT-G block, most of the essential commands are enumerated in 'Operation'.

I2C Registers:

The NXTCam appears as a set of few registers as follows.

Register	Read	Write	Comments
0x00-0x07	Software version – *(Vn.nn)*	-	
0x08-0x0f	Vendor Id – *mndsnsrs*	-	
0x10-0x17	Device ID – *NXTCam*	-	
0x41	-	Command	This register is command register. A command written here will be executed.
0x42	Number of objects detected	-	Shows how many objects are being tracked. Zero indicates that there are no objects being tracked.
0x43	1st object color	-	This is the first object color as per the sorting method selected.
0x44[1]	1st object - X upper left		Upper left X coordinate of first object
0x45	1st object - Y upper left		Upper left Y coordinate of first object
0x46	1st object - X lower right		Lower right X coordinate of first object
0x47[2]	1st object - Y lower right		Lower right Y coordinate of first object
0x48	2nd object color		
0x49-0x4C	2nd object co-ordinates		
0x4D	3rd object color		
0x4E-0x51	3rd object co-ordinates		
0x52	4th object color		
0x53-0x56	4th object co-ordinates		
0x57	5th object color		

[1] In case of line tracking mode, these are coordinates of beginning and end points of the line.

[2] This repeats for all 8 objects. Please note that object position and coordinate are overwritten if new object is detected, otherwise previous value is retained.

Register	Read	Write	Comments
0x58-0x5B	5th object co-ordinates		
0x5C	6th object color		
0x5D-0x60	6th object co-ordinates		
0x61	7th object color		
0x62-0x65	7th object co-ordinates		
0x66	8th object color		
0x67-0x6A	8th object co-ordinates		
0x6B	No. of registers to Read	No. of registers to Write	This is the number of registers you need to read or write from NXTCam image sensor
0x6C	1st image sensor register Address	1st image sensor register Address	
0x6D[3]	1st image sensor register Data	1st image sensor register Data	1st register Data read from image sensor or written to image sensor
……	……	……	
0x7A	8th image sensor register Address	8th image sensor register Address	
0x7B	8th image sensor register Data	8th image sensor register Data	
0x80[4]	Color map data Red 0	Color map data Red 0	0x80 – 0xAF - These registers are used for Colormap data reading and writing
0x80	Color map data Red 0	Color map data Red 0	
0x81	Color map data Red 1	Color map data Red 1	
0x82	Color map data Red 2	Color map data Red 2	
0x83	Color map data Red 3	Color map data Red 3	
0x84	Color map data Red 4	Color map data Red 4	
0x85	Color map data Red 5	Color map data Red 5	
0x86	Color map data Red 6	Color map data Red 6	
0x87	Color map data Red 7	Color map data Red 7	

[3] If you need to read image sensor register 0x00 (i.e. 1 register) then follow this:
Write 0x01 to register 0x6B, Write 0x00 to register 0x6C
Run command 'H'
Results will be stored in register 0x6D
If you need to write to image sensor register 0x00 (i.e. 1 register) then follow this:
Write 0x01 to register 0x6B, Write 0x00 to register 0x6C, Write data to register 0x6D
Run command 'C'

[4] If you need to read Colormap data from NXTCam engine, then follow this:
Run command 'G'
Read registers 0x80 – 0xAF.
If you need to write Colormap data to NXTCam engine, then follow this:
Write your color map data in registers 0x80 – 0xAF,
Run command 'S'

Register	Read	Write	Comments
.....	.....	.....	
0x8F	Color map data Red 15	Color map data Red 15	
0x90	Color map data Green 0	Color map data Green 0	
0x91	Color map data Green 1	Color map data Green 1	
.....	.....	.....	
0x9F	Color map data Green 15	Color map data Green 15	
0xA0	Color map data Blue 0	Color map data Blue 0	
0xA1	Color map data Blue 1	Color map data Blue 1	
.....	.....	.....	
0xAF	Color map data Blue 15	Color map data Blue 15	

Changing the I2C Address

The factory default I2C address of NXTCam is 0x02.

This address can be changed. To set an address different from default address, send sequence of following commands on the command register:

0xA0, 0xAA, 0xA5, <new I2C address>

Note: Send these commands with no break/read operation in between. This new address is effective immediately. Please note down your new address for future reference.

Alternately, you can download device scan and address change programs from following URLs, and change them to suit your needs:

EV3/NXT:

http://www.mindsensors.com/blog/how-to/change-i2c-device-address

PiStorms:

http://www.mindsensors.com/blog/how-to/change-i2c-device-address-with-pistorms

Appendix B: NXTCAM Colormap Structure

The colormap is a 48 byte buffer in NXTCam memory.

From this buffer each of RGB color is assigned 16 bytes for that color.

i.e. first 16 bytes are for Red, next 16 for green and remaining 16 for blue.

These 16 bytes store matching preference for that color's absolute value ranges as

0-16-32-48-64-80-96-112-128-144-160-176-192-208-224-240-255

i.e. first byte is set for color range 0 to 15, second byte for range 16 to 31, third byte for 32 to 48 and so on...

Each bit in the byte contains mask for each object (i.e. 8 objects) which is set to 1 if you want matching object.

RGB values for all 8 Colors are described (and stored as follows).

For example the blue pen seen in the picture above has color RGB value ranges as follows:

Red: 0-64

Green: 0-64

Blue: 128-240

The values to store this color in 'Color 3' is shown in table below.

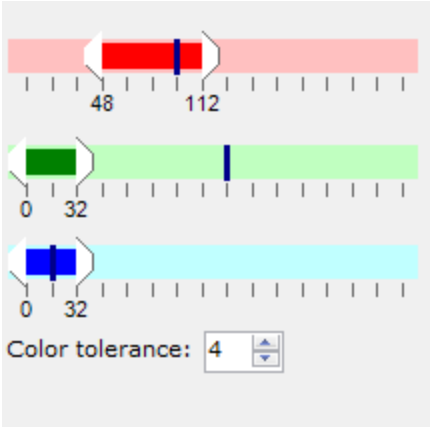
Whereas Red pen has RGB value ranges as in adjacent picture:

Red: 48-112

Green: 0-32

Blue: 0-32

This color is stored in 'Color 2' in table below:



Colormap table: There are 48 rows in this table, corresponding to 48 bytes. The bits of each byte are shown in cells of corresponding row. 'X' indicates the bit is turned ON in corresponding byte.

		Color 1	Color 2	Color 3	Color 4	Color 5	Color 6	Color 7	Color 8
Byte 1	Red 0-15			X					
Byte 2	Red 16-31			X					
Byte 3	Red 32-47			X					
Byte 4	Red 48-63		X	X					
Byte 5	Red 64-79		X						
Byte 6	Red 80-95		X						
Byte 7	Red 96-111		X						
Byte 8	Red 112-127								
Byte 9	Red 128-143								
Byte 10	Red 144-159								
Byte 11	Red 160-175								
Byte 12	Red 176-191								
Byte 13	Red 192-207								
Byte 14	Red 208-223								
Byte 15	Red 224-239								
Byte 16	Red 240-255								
Byte 17	Green 0-15		X	X					
Byte 18	Green 16-31		X	X					
Byte 19	Green 32-47			X					
Byte 20	Green 48-63			X					
Byte 21	Green 64-79								
Byte 22	Green 80-95								
Byte 23	Green 96-111								
Byte 24	Green 112-127								
Byte 25	Green 128-143								
Byte 26	Green 144-159								
Byte 27	Green 160-175								
Byte 28	Green 176-191								
Byte 29	Green 192-207								
Byte 30	Green 208-223								
Byte 31	Green 224-239								
Byte 32	Green 240-255								
Byte 33	Blue 0-15		X						
Byte 34	Blue 16-31		X						
Byte 35	Blue 32-47								
Byte 36	Blue 48-63								
Byte 37	Blue 64-79								
Byte 38	Blue 80-95								
Byte 39	Blue 96-111								
Byte 40	Blue 112-127								
Byte 41	Blue 128-143			X					
Byte 42	Blue 144-159			X					
Byte 43	Blue 160-175			X					
Byte 44	Blue 176-191			X					
Byte 45	Blue 192-207			X					
Byte 46	Blue 208-223			X					
Byte 47	Blue 224-239			X					
Byte 48	Blue 240-255			X					

Library Functions for reading/writing Colormap Structure

It is advisable to view colors and update the Colormap using NXTCamView software. However if you must do this through your program, the NXC library at following URL has functions and sample programs to update Colormap. Download the Colormap - read/write programs and modify them to suite your needs.

http://www.mindsensors.com/index.php?controller=attachment&id_attachment=94