

Basler Components



Using IP Camera Live Buffers and Alarm Buffers

APPLICATION NOTES

Document Number: AW000810

Version: 01 Language: 000 (English)

Release Date: 23 March 2009

Contacting Basler Support Worldwide

Europe:

Basler AG
An der Strusbek 60 - 62
22926 Ahrensburg
Germany
bc.support.ip@baslerweb.com

Americas:

Basler, Inc.
855 Springdale Drive, Suite 160
Exton, PA 19341
U.S.A.
bc.support.ip@baslerweb.com

Asia:

Basler Asia Pte. Ltd
8 Boon Lay Way
03 - 03 Tradehub 21
Singapore 609964
bc.support.ip@baslerweb.com

www.basler-ipcam.com

1 Introduction

The IP Camera can provide up to three video streams simultaneously. In brief, this means, that you can specify up to three different rectangular regions of the camera's sensor, and you can assign each region to an individually configured video stream ("individually" not only means that each stream shows a different field of view, it also means that each stream can use a different encoder, image quality, bit rate, frame rate, etc.).

When the camera is first powered on, at least one video stream, stream 0, becomes immediately active and can not be switched off. The other two streams, stream 1 and stream 2, can be enabled or disabled as you desire.

Even if the camera is not connected to a network, it immediately begins capturing images and storing them in the camera's internal buffers. It is important to know, that each of the three available video streams has two dedicated image buffers, a Live Buffer and an Alarm Buffer. As soon as a video stream starts (e.g. stream 0, which will be automatically started when the camera is powered on), the camera begins placing the captured images into the live buffer for the stream. If the alarm buffer for the stream is properly configured and is "armed", the same images will be also placed into the alarm buffer.

The size of each buffer is configurable and can range from 2048 to 57856 KB. The buffers are implemented as ring buffers. This means that once the buffer is full, as newer images are placed into the buffer they will begin overwriting the oldest images. As shown in Figure 1 on [page 2](#), when the camera captures image N and transfers it via stream 0 to the buffers, image N is written into the current position of the live and the alarm buffer. Once this is done, the buffers rotate by one image position. The old image at this position will now be overwritten by the next image captured.

In Figure 1, the live buffer and the alarm buffer are only shown for stream 0, but of course, stream 1 and stream 2 both have their own live buffer and alarm buffer as well.

As mentioned earlier, the writing of image data into the buffers happens totally independently from the transmission of the image data over the Ethernet. Even if the camera wasn't connected to a network at all, it would still capture images and store them in the ring buffers as long as the camera is powered on.

The number of images that can fit into a ring buffer depends on the size of the buffer and the size of the compressed images. On the IP Camera, the size of the buffers on each stream is configurable, as is the size of the images in the stream. The alarm buffer on each stream can also be disabled if desired.

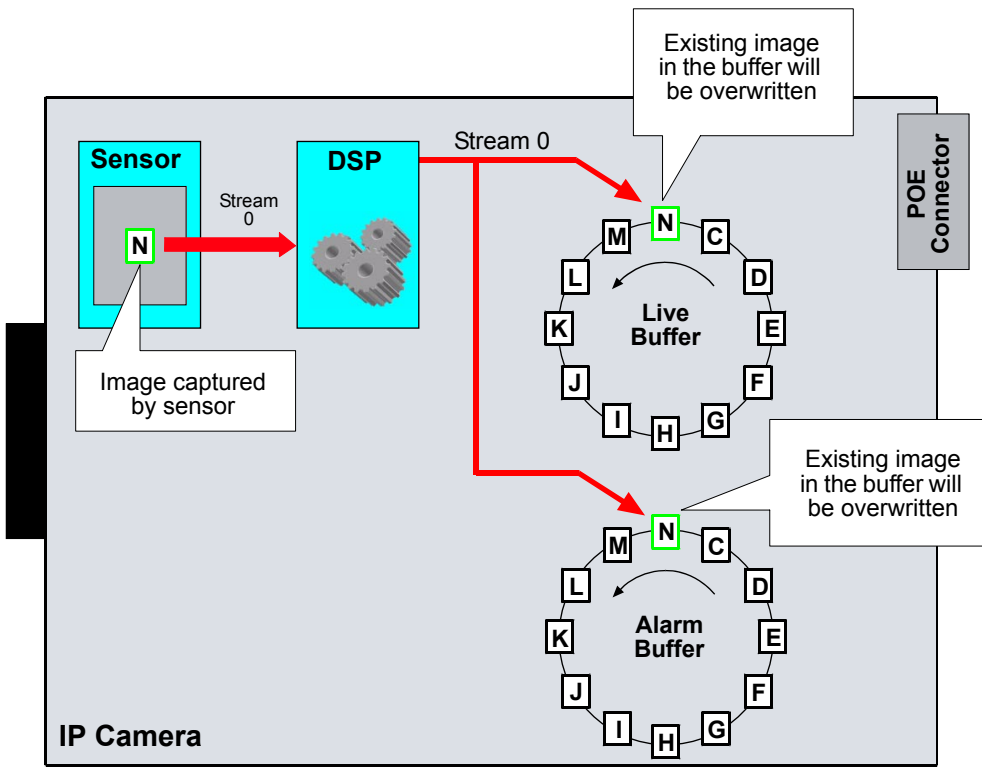


Fig. 1: Buffers on Stream 0

2 A Closer Look at a Live Buffer

If a client requests access to a video stream (for example, to stream 0), by default, the client would access the content of the live buffer at the position where the most recent images are entering the buffer (image N in Figure 2).

This request:

```
http://<camera>/cgi-bin/mjpeg?stream=0
```

would yield a stream that includes the most current images as shown in Figure 2.

However the client can also access the image data at any position of the ring buffer. This means that the client can "look back into history" and get images that were captured some time ago. This is done by issuing a request that includes the "timeshift" and "seek" parameters

This request:

```
http://<camera>/cgi-bin/mjpeg?buffer=0&mode=timeshift&seek=2000
```

would yield a stream that includes images captured 2000 ms ago as shown in Figure 3 on [page 4](#).

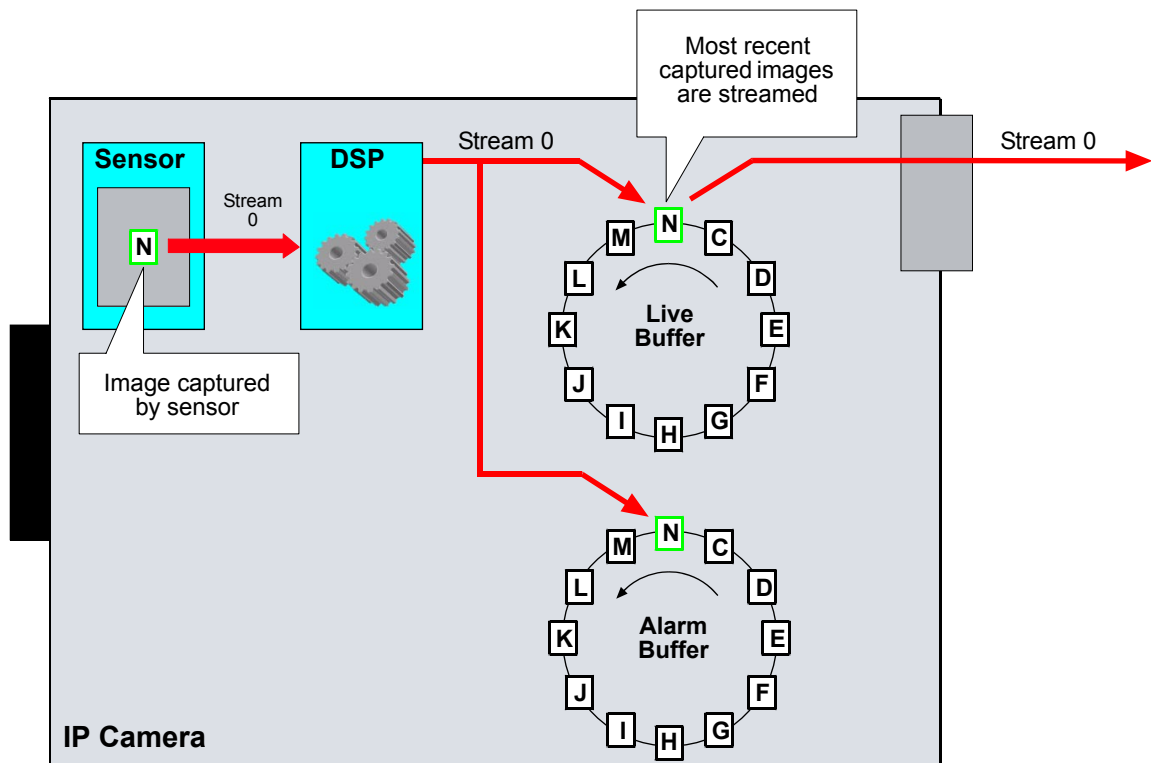


Fig. 2: A Live Stream with the Most Current Images

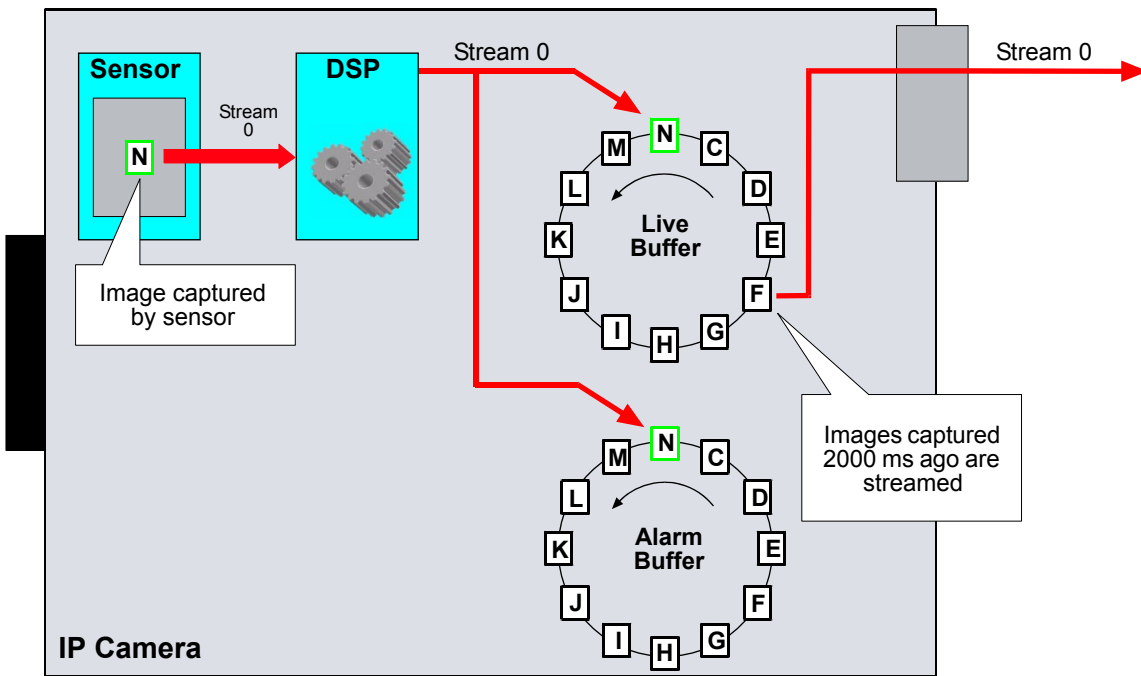


Fig. 3: A Live Stream with Images Captured in the Past

3 A Closer Look at an Alarm Buffer

When the alarm buffer on a stream is enabled and "armed", the captured images are placed into the alarm buffer in the same manner as they are in a live buffer. Typically, the alarm buffer on a stream is configured for a larger size than the live buffer, so that the alarm buffer can hold more images.

A client can request access to a video stream from the alarm buffer at the position where the most recent images are entering the buffer (image N in Figure 4).

This request:

```
http://<camera>/cgi-bin/mjpeg?buffer=1&mode=live
```

would yield a stream that includes the most current images from the alarm buffer as shown in Figure 4.

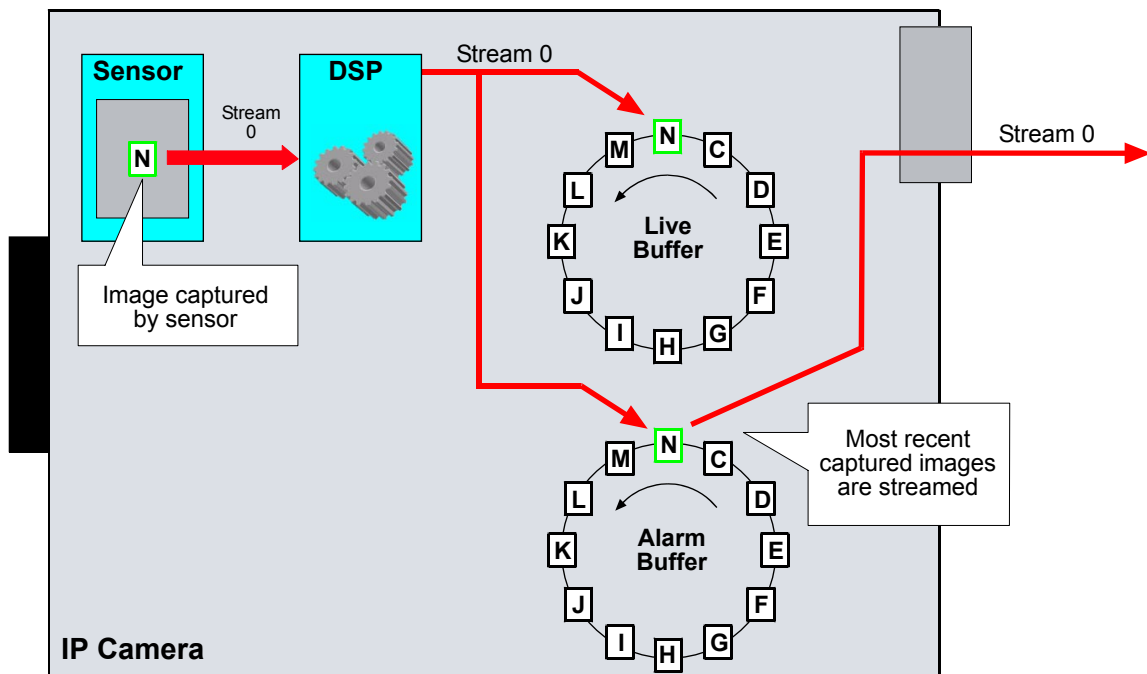


Fig. 4: An Alarm Stream with the Most Current Images

The point at which an alarm buffer begins to work differently than a live buffer is when an alarm condition is triggered. When an alarm condition is triggered (for example, by the camera's built-in motion detection function or by an external signal applied to the camera's digital input), the camera will STOP putting images into the alarm buffer at a certain point. This "certain point" can be configured by setting the "Post Alarm Buffer Size" parameter. When you configure an alarm buffer, you must configure both the total size of the buffer and the size of the portion of this buffer that will be used as "post alarm". The post alarm part of the alarm buffer is reserved for images that are taken immediately AFTER an alarm occurs.

As an example, assume that the Alarm Buffer Size parameter is set to a value that will allow the alarm buffer to hold a total of 12 images (in reality, the storage capacity is much bigger). Also assume that the Post Alarm Buffer Size parameter is set to a value that will allow the post-alarm portion of the alarm buffer to hold nine images. Finally assume that the buffer has filled with images and is currently overwriting the oldest images with the newest images as you would normally expect it to do.

If an alarm is now triggered, the camera will continue to put images into the alarm buffer until the post-alarm portion of the buffer is filled. At this point the camera will STOP putting any images into the alarm buffer. The result is that the buffer now contains three images that were captured before the alarm was triggered and nine images that were captured after the alarm was triggered as shown in Figure 5. When the buffer is accumulating post alarm images, it is said to be in an "active" state. And when it has stopped and is filled with pre and post alarm images, it is said to be in a "done" state.

The buffer will remain stopped and the images will remain in the alarm buffer until the alarm buffer has been rearmed.

And all of this can happen without causing any network load!

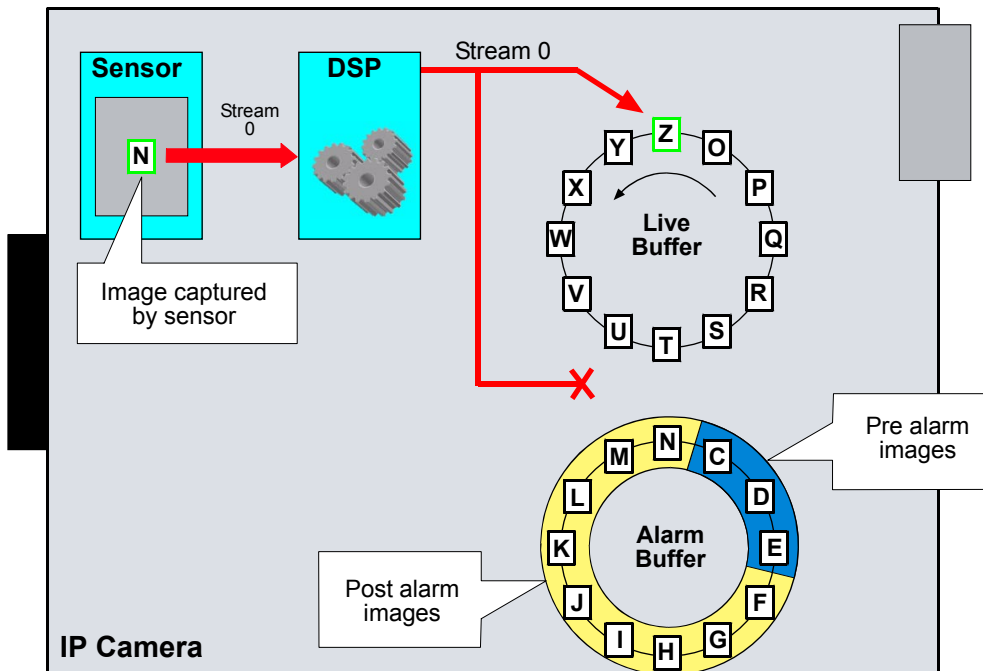


Fig. 5: An Alarm Buffer in the Done State

When the alarm buffer is in the done state, a client can request access to a video stream from the buffer. This request:

```
http://<camera>/cgi-bin/mjpeg?buffer=1&mode=replay&fps=5.0
```

would yield a stream that includes the images held in the alarm buffer. The stream would send the images in the alarm buffer at a rate of five frames per second and would stop when all of the images in the buffer have been streamed.

Use Case:

A Supervisory Camera Using No Network Bandwidth and No Disk Space

You can configure the camera so that it will trigger an alarm condition if motion is detected. You can also configure the camera so that an alarm buffer is enabled, the buffer has both pre and post alarm space allocated in it, and the buffer is armed.

In this situation, the camera will not need to transmit data to any client. This means that the camera will not use any network bandwidth and that no data needs to be stored on a hard disk.

Nevertheless, the camera is already doing its job - it is supervising the field of view and waiting to detect motion.

If an alarm is triggered, the camera can actively inform a piece of client software (NVR/VMS) by pushing an http comment to the client. The client could then, for example, simply request the camera to stream the content of the alarm buffer because this is the only information that the client is actually interested in. This means that only relevant alarm data is actually using network bandwidth and requiring disk storage.

Revision History

Doc. ID Number	Date	Changes
AW00081001000	23 Mar 2009	Initial release of this document.

