

Справочное руководство

DiscretePhoton H.264 encoder

Справочное руководство

О DiscretePhoton H.264 encoder

DiscretePhoton H.264 кодер для Windows версии предоставляется как DirectX Media Object (DMO) в 32-битной или 64-разрядного двоичного. Это может быть использован непосредственно, либо в среде DirectShow.

DiscretePhoton H.264 encoder поддерживает до 64 одновременных threads. Однако фактическое число потоков определяется числом CPU-cores (или hyper-threads) вашей системы и размера кадра. Можно только уменьшить его.

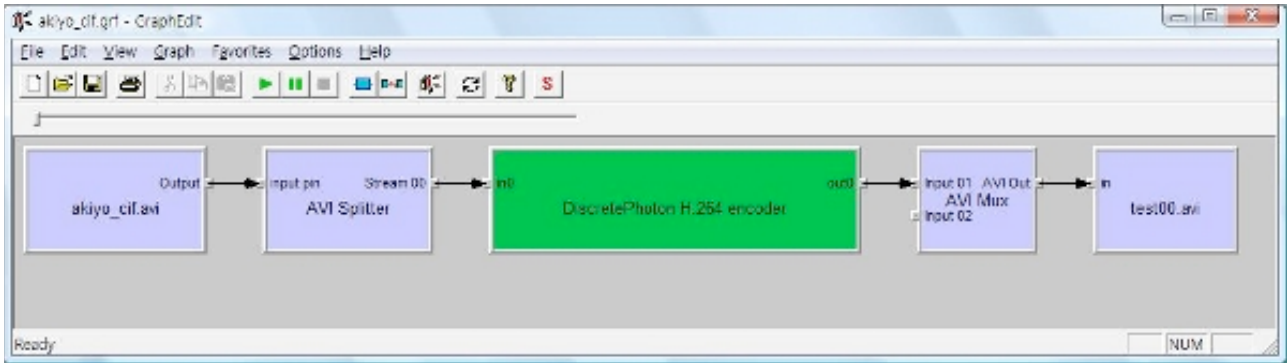
Мультипоточность DiscretePhoton H.264 encoder основана на wavefront. Все рабочие потоки сосредоточены на кодировании экрана самого последнего ввода, что может привести к очень низкой латентности кодирования без какого-либо снижения окончательного качества кодирования или уменьшения величины сжатия. Также CBR rate-control данного продукта настроен на низкую латентность. Таким образом, DiscretePhoton H.264 кодер очень хорошо приспособлены для таких приложений, как видеоконференции, а также других типов приложений.

Низкая латентность кодирования также означает низкую латентность вариации, что может привести к низкому уровню frame-drop rate при кодировании в реальном времени.

Некоторые показатели производительности DiscretePhoton H.264 encoder могут быть найдены на сайте: [7-th annual video codec comparison by MSU](#).

Для получения дополнительной информации и скачивания демонстрационных версий, посетите www.discretephoton.com.

Использование GraphEdt.exe



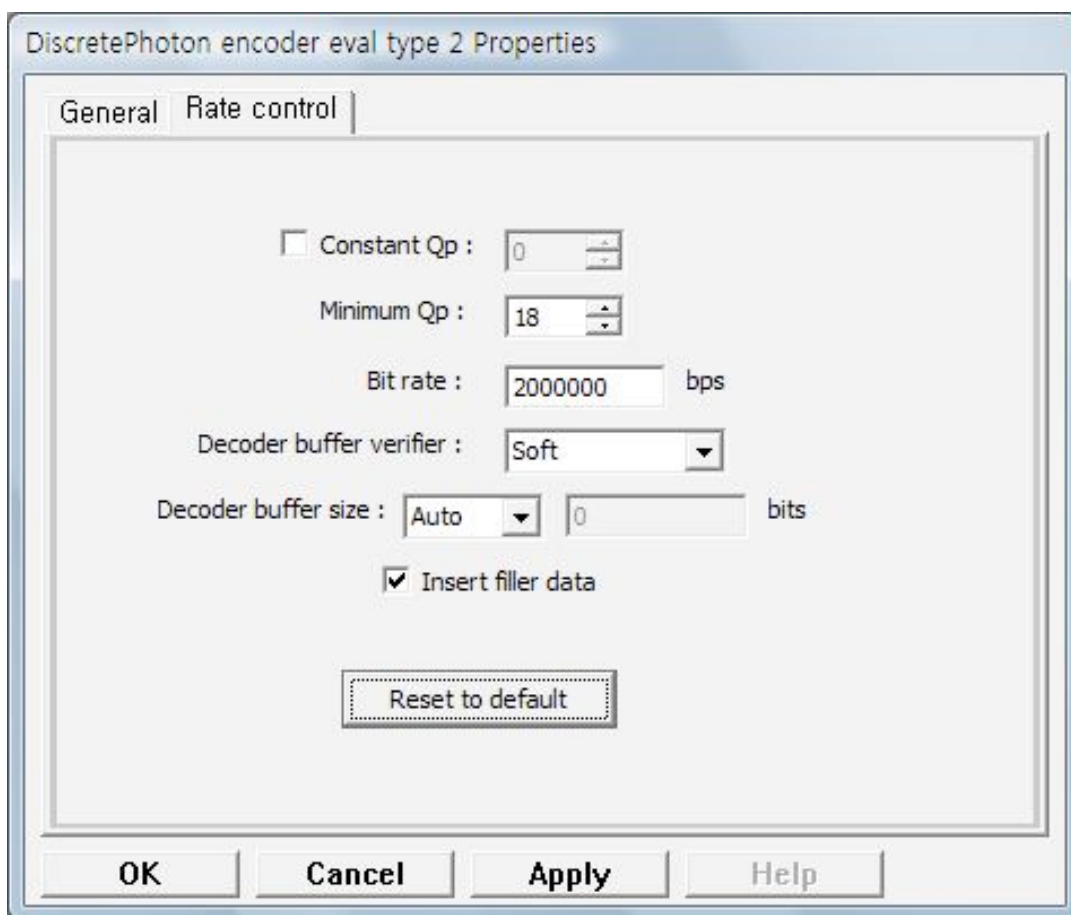
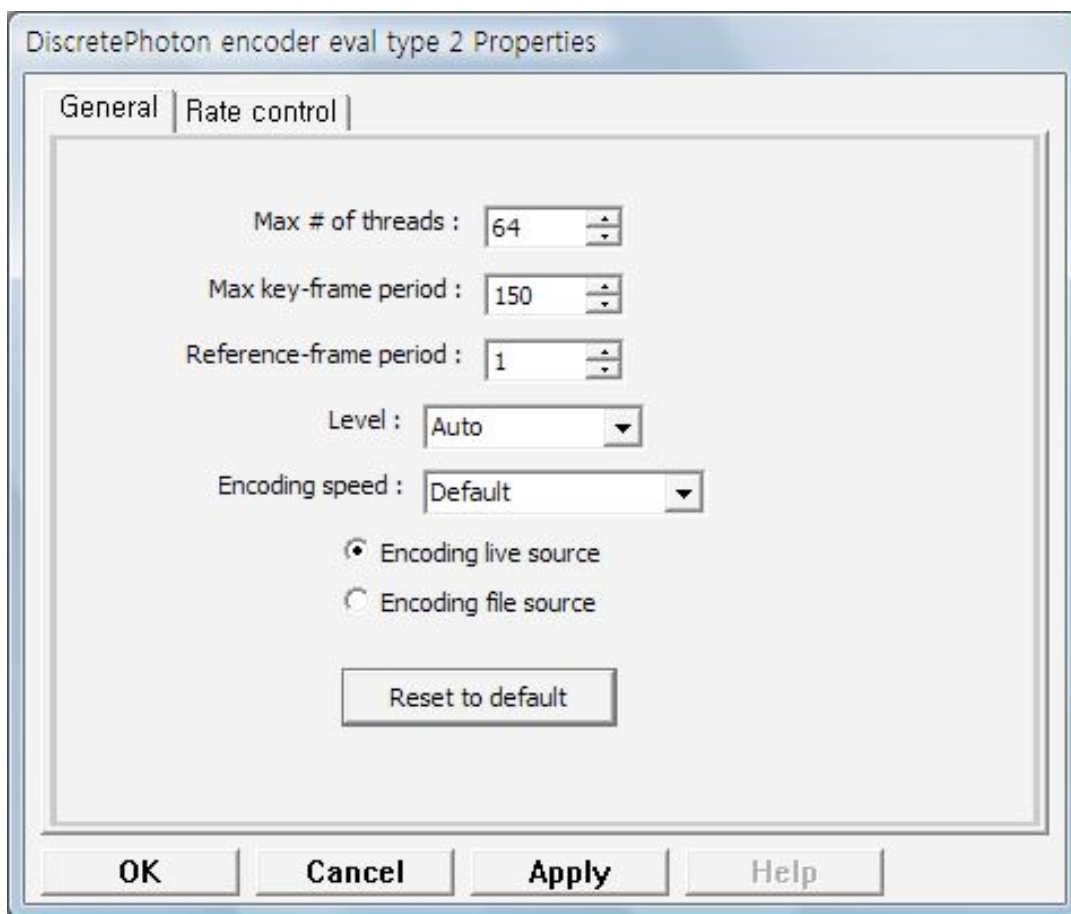
GraphEdt.exe содержится в Windows SDK. Вы можете использовать его для построения и тестирования графа фильтра DirectShow визуально.

После инсталляции 32-битной версии DiscretePhoton H.264 encoder Вы можете найти его в GraphEdt.exe на Graph -> Insert Filters... -> Video Compressors в строке меню.

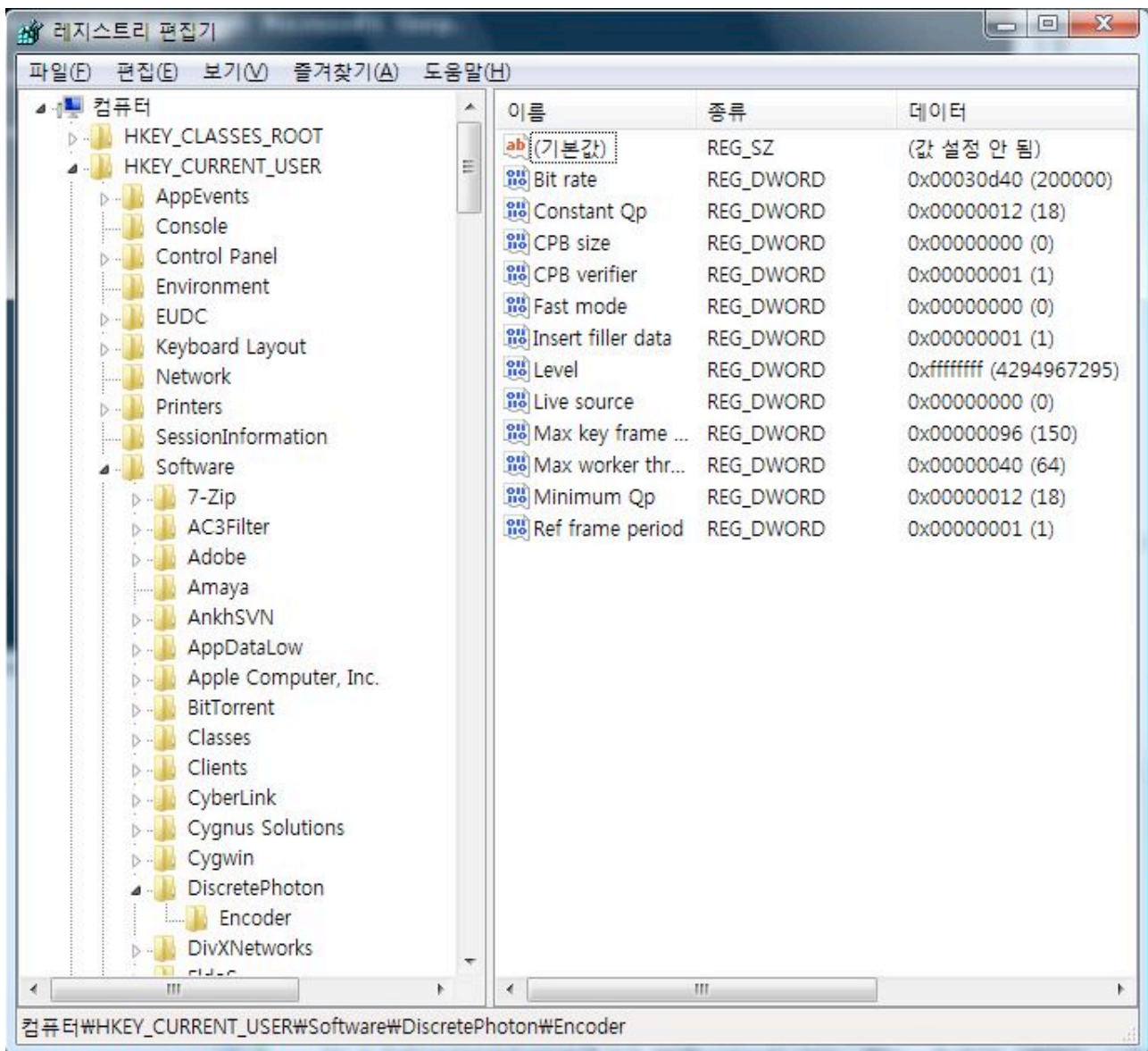
Зеленый цвет DiscretePhoton H.264 encoder узел показывает, что это на самом деле не DirectShow фильтр, но DMO (DirectX Media Object) с обертыванием DirectShow.

DiscretePhoton H.264 encoder поддерживает такие вводные видео форматы: **I420, YV12, UYVY, YUYV, RGB24, RGB32**. А это выводит H.264 byte stream (FourCC: **H264**).

Если щелкнуть правой кнопкой на узле DiscretePhoton H.264 encoder, страницы свойств будет показано, как ниже.



Есть две вкладки 'General' и 'Rate Control'. После изменения любого значения и применить его, измененные значения хранятся в реестре Windows(HKEY_CURRENT_USER\Software\DiscretePhoton\Encoder). Пользователь может читать или записывать эти значения реестра в программе пользователя.



Значение каждого поля следующее:

Max # of threads: (название в реестре: "Max worker threads") Максимальное число потоков, применяемых во время кодирования. Это число является не фактическим, а максимальным, потому что фактическое число определяется числом CPU-cores (или hyper-threads) ПК пользователя и размером экрана. Если пользователь не хочет использовать 100% CPU в процессе кодирования, то он может уменьшить это число ниже, чем число CPU-ядер.

Max key-frame period: (название в реестре: "Max key frame period") Здесь ключевой кадр означает IDR-кадр H.264. Число этого тоже является не фактическим, а максимальным, потому что фактические ключевые кадры могут появляться чаще,

чем значения, определенные для эффективности кодирования. Минимальная частота ключевых кадров составляет половину от значения, определенного здесь.

Reference-frame period: (название в реестре: "Ref frame period") По умолчанию каждый кадр будет использоваться в качестве референсного кадра. Однако за счет увеличения этого значения(в этом случае только часть кадров будет использована в качестве референсного кадра) и уменьшения значения максимальной частоты ключевого кадра, может быть достигнута отказоустойчивость во время передачи видеосигнала по ненадежной сети(например, UDP).

Level: (значения в реестре: -1:Auto, 0:Level_1, 1:Level_1b, ...) Это кодирования H.264 уровне (как указано в приложении A.3 спецификации). Как правило, вы можете установить его в качестве 'Авто'. Тогда уровень будет определяться автоматически с другими параметрами и видеовхода.

Encoding speed: (название в реестре: "Fast mode". 0:Default, 1:Fast, 2:Even faster) Более быстрая скорость кодирования может быть достигнута, немного жертвуя качеством видео. Следовательно, если пользователь использует достаточно высокую выходную скорость передачи битов, он может настроить это в режиме 'Fast' или 'Even faster'. Но если пользователь использует низкую выходную скорость передачи битов, то режим 'Even faster' может привести к образованию блока на экране.

Encoding live source / Encoding file source: (название в реестре: "Live source". 0:file source, 1:live source) В случае кодирования 'живого' видео с веб-камеры, ТВ-тюнера и т.д., выберите 'Encoding live source', а в случае кодирования файлов, выберите 'Encoding file source'.

Constant Qp: (значения в реестре: 0 деактивизирует это) В случае применения этого пункта качество выходного видео будет почти постоянным, а выходная скорость передачи битов может сильно колебаться. В данном случае процесс регулировки выходной скорости передачи битов будет исключена. Следовательно, рекомендуется не использовать данный пункт для видео, которое передается по сети. Действительный диапазон значений – 18 ~ 51. Большое значение означает высокое соотношение сжатия и низкое качество.

Minimum Qp: Значения Qp(параметр квантования) ниже, чем значения, определенные здесь не будут применяться во время контроля выходной скорости передачи битов.

Bit rate: Выходная скорость передачи битов. Число битов в секунду

Decoder buffer verifier: (название в реестре: "CPB verifier". 0:Hard, 1:Soft, 2:None) Это настраивает жесткость процесса подтверждения буфера декодера. Здесь буфер декодера соответствует буферу кодированного изображения(CPB), приведенному в приложении C Спецификации H.264 (Внимание: даже 'Hard' не означает совершенное подтверждение).

Decoder buffer size: (название в реестре: "CPB size") Размер в битах. При настройке этого значения в режиме 'Auto', настраивается как максимальное допустимое значение(VCL HRD).

Insert filler data: Если буфер декодера достиг вблизи overflow, а также значение Qp достигло минимального, то filler data может быть вложено в выходной поток битов для поддержания выходного скорости передачи битов. Как правило, данный пункт должен быть настроен.

Программирование с применением DirectShow

Я считаю, Есть несколько хороших ссылок о программировании в DirectShow окружающей среды.

Вы также можете скачать простые примеры программ, которые используют DiscretePhoton H.264 кодер от:

32-битная версия:

<http://www.discretephoton.com/php/downloader.php?f=Examples.zip>.

64-битная версия:

http://www.discretephoton.com/php/downloader.php?f=Examples_x64.zip.

Для создания версии C++ требуется DirectShow base class library, который содержащиеся в Windows SDK. И C # версия требует DirectShow.Net. Более подробную информацию Вы можете найти в README.htm.