Technologie odbioru danych 1GigE

Wprowadzenie

Interfejs 1GigE oferuje ograniczoną maksymalną przepustowość, a zastosowana technologia odbioru danych z kamery (driver producenta) dodatkowo wpływa na czas w jakim obraz stanie się dostępny do obróbki przez procesor.

Spis treści

WPROWADZENIE **1**
WSTĘP 2
KONFIGURACJA INTERFEJSU STANDARDOWA VS MACHINE VISION . 2
Strumień danych kamer FullHD na 1GigE 2
Technologie transferu danych 3
- Konfiguracja Jumbo Frames (Warstwa Fizyczna/Karty Sieciowej) 3
- Filter Drivers (Sterowniki Filtrujące) vs. Generyczny Stos OS 3
- Mechanizm Zero-Copy i Ring Buffers (Kolejki Kołowe) 3
- Socket Pooling i Hardware Interrupts 4

PODSUMOWANIE EFEKTÓW 4

© A1Pixel sp z o.o. ul. Krasińskiego 15/3, 87-100Toruń, Polska, www.a1pixel.pl ,

Wstęp

Standard 1GigE jest klasycznym, niezwykle popularnym, używanym dla kamer o rozdzielczości do 5 MP. Na możliwości systemów wizyjnych opartych o kamery 1GigE silnie wpływa driver kamery i konfiguracja łącza.

Konfiguracja interfejsu. Standard vs Machine Vision .

Strumień danych kamer FullHD na 1GigE

Interfejs 1GigE oferuje maksymalną przepustowość surową na poziomie 1 Gbps (125 MB/s). W praktyce, po odliczeniu narzutu protokołów sieciowych (interramki, nagłówki IP/UDP/GVSP), realne pasmo użytkowe na dane obrazowe wynosi ok. 115 MB/s.

Obciążenie interfejsu np. przez kamerę 2.3 MP (np. sensor Sony IMX249) o rozdzielczości 1920 x 1200 pikseli (Minimalnie wiecej niż FullHD 1920x1080):

- Format Mono8 (8-bit / 1 bajt na piksel):
    - Jedna klatka = 1920 x 1200 x 1 bajt = approx 2,30 MB .
    - Przy 40 Hz (klatkach na sekundę): 40 x 2,30 MB = 92 MB/s Bezpieczny margines, ok. 80% pasma 1GigE .
    - Przy 60 Hz: 60 x 2,30 MB = 138 MB/s Przekracza możliwości 1GigE (nastąpi gubienie pakietów).
- Format Bayer/RGB24 (24-bit / 3 bajty na piksel):
    - Jedna klatka = 1920 x 1200 x 3 x bajty = approx 6,91 MB/s.
    - Maksymalny klatkaż limitowany pasmem to zaledwie ~16 Hz 16 x 6,91 = 110,5 MB/s

Aby uzyskać niskie opóźnienie i wysoki klatkaż na 1GigE, kamera musi przesyłać obraz w formacie surowym (Mono8 lub BayerGR8). Konwersja do pełnego koloru (debayering/debayering) musi zostać oddelegowana na stronę odbiorczą (CPU/GPU komputera), aby nie marnować pasma sieciowego.

Technologie transferu danych

Gdy kamera wysyła klatkę o rozmiarze 2.3 MB, nie wysyła jej w jednym bloku. Standardowy pakiet sieciowy Ethernet (MTU) ma 1500 bajtów. Oznacza to, że jedna klatka obrazu zostaje poszatkowana na około 1500–1600 mniejszych pakietów UDP.

Zadaniem strony odbiorczej jest odebranie tych pakietów z gigantyczną prędkością, ułożenie ich w spójną klatkę w pamięci RAM i powiadomienie aplikacji o gotowości obrazu, generując przy tym jak najmniejsze opóźnienie. Odbywa się to poprzez następujące technologie:

Konfiguracja Jumbo Frames (Warstwa Fizyczna/Karty Sieciowej)

To pierwsza linia obrony przed opóźnieniami. Standardowe pakiety (MTU 1500) zmuszają kartę sieciową i procesor komputera do obsłużenia 1500 przerwań na każdą klatkę obrazu.

- Włączenie Jumbo Frames (MTU 9000) po stronie kamery i karty sieciowej komputera pozwala powiększyć pakiety sześciokrotnie.
- Liczba pakietów na jedną klatkę 2.3 MP spada z 1500 do około 260.
- Efekt: Obciążenie procesora przerwaniami (Interrupt Rate) spada o 80%, co drastycznie zmniejsza jitter i ryzyko utraty pojedynczego pakietu (która unieważniłaby całą klatkę).

Filter Drivers (Sterowniki Filtrujące) vs. Generyczny Stos OS

Standardowy stos sieciowy systemu operacyjnego (np. Windows NDIS) traktuje pakiety z kamery jak zwykły ruch internetowy. Pakiet przechodzi przez firewall, warstwę TCP/IP, bufory systemowe, i jest wielokrotnie kopiowany w pamięci. Dla systemu wizyjnego oznacza to losowe opóźnienia od 5 do 15 ms.

Dlatego producenci kamer maszynowych dostarczają dedykowane sterowniki:

- Filter Driver / Performance Driver: Przechwytuje pakiety protokołu GigE Vision (GVSP - GigE Vision Streaming Protocol) na najniższym możliwym poziomie – tuż nad warstwą sprzętową karty sieciowej (NIC).
- Omija on standardowy, ciężki stos sieciowy systemu operacyjnego. Pakiety są natychmiast kierowane do pamięci dedykowanej dla aplikacji wizyjnej.

Mechanizm Zero-Copy i Ring Buffers (Kolejki Kołowe)

W klasycznym programowaniu dane z karty sieciowej trafiają do pamięci jądra (*Kernel Space*), skąd aplikacja musi je skopiować do swojej pamięci (*User Space*). Kopiowanie 2.3 MB kilkadziesiąt razy na sekundę marnuje cenne cykle procesora.

Nowoczesne API odbiorcze używają technologii Zero-Copy:

1. Aplikacja alokuje w pamięci RAM tzw. Ring Buffer (kolejkę kołową składającą się zazwyczaj z 3 do 10 buforów o rozmiarze klatki, czyli ok. 2.5 MB każdy).
2. Sterownik filtrujący (Filter Driver) otrzymuje adresy tych buforów i instruuje kartę sieciową, aby składała pakiety bezpośrednio w tych obszarach pamięci RAM.
3. Gdy ostatni pakiet danej klatki (oznaczony flagą *End-of-Frame*) zostanie odebrany, sterownik natychmiast przekazuje aplikacji wskaźnik (*pointer*) do tego miejsca w pamięci. Dane nie są kopiowane ani razu.

Socket Pooling i Hardware Interrupts

Komputery przemysłowe wykorzystują zaawansowane mechanizmy sterowania procesami:

- Polling (Odpytywanie): Zamiast czekać na przerwanie ze sterownika o nadejściu każdego pakietu, wątek odbiorczy w aplikacji działa w pętli o wysokim priorytecie i stale sprawdza stan rejestrów pamięci (redukuje to opóźnienie wywołane wybudzaniem rdzenia procesora ze stanów oszczędzania energii).
- Affinity (Spójność rdzeni): W systemach czasu rzeczywistego dedykuje się jeden konkretny rdzeń procesora (np. Rdzeń 3) wyłącznie do obsługi karty sieciowej GigE i wątku odbiorczego. Dzięki temu przetwarzanie obrazu nie jest przerywane przez inne procesy systemu Windows/Linux.

Podsumowanie efektów

Zastosowanie powyższych technologii drastycznie skraca czas potrzebny na samo odebranie obrazu 2–3 MP:

Konfiguracja strony odbiorczej	**Czas składania i transferu klatki do RAM aplikacji**	**Stabilność (Jitter)**	**Obciążenie CPU (Intel Core i5)**
Standardowa (MTU 1500, Domyślny driver Windows)	**8 – 20 ms**	Bardzo niestabilny 5ms	~25-30% (marnowane na obsługę przerwań)
Zoptymalizowana (MTU 9000, Filter Driver, Zero-Copy, Core Affinity)	**~1.2 – 2.5 ms** (bliskie fizycznemu limitowi kabla)	Ekstremalnie stabilny 0.1ms	~2-5%