



10GBASE-Tに対応!

# TOE 10G

## IPcore

**TCP Offloading Engine IP Core**

## 最高速のPCインターフェイスを実現

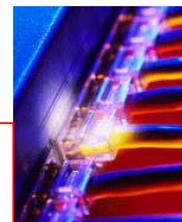
## アジェンダ

- ・ 10GbE&TCP/IP概要
  - 10GbEでのTCP/IP実装メリット・デメリット
- ・ TOE10G-IPコアの概要
- ・ コアの動作
  - 初期化
  - 高速送信
  - 高速受信
- ・ ユーザI/F・バッファ容量のパラメタライズ
- ・ リファレンス・デザイン
- ・ リソース例・実機パフォーマンス例



# 10GbE概要

- 10GbE (10ギガビット・イーサネット)とは
  - 業界標準の高速ネットワーク・インターフェイス
  - 10Gbpsの転送速度
  - 多様な接続方法が混在する



全ての接続方法で  
TOE10G-IPの適用が可能

| 接続方法      | 長所                   | 短所                | 備考                 |
|-----------|----------------------|-------------------|--------------------|
| 光ケーブル     | 低レイテンシ(100ns)<br>長距離 | 高価                | 光モジュール+<br>ケーブルが必要 |
| ダイレクトアタッチ | 低価格                  | 短距離(最大5-<br>7m程度) | SFP+ソケットに<br>直接挿入  |
| 10GBASE-T | 低価格,<br>汎用(RJ-45)    | 高レイテンシ<br>(2us)   | 特殊(LDPC)な符<br>号化方式 |

## 10GbEの主な接続方法

# 10GbEでのTCP/IP実装メリット

- TCP/IPの実装メリット
  - イーサネットの標準プロトコル規格
  - データの欠損がなく転送データの信頼性が確保される
  - 多くのOSが標準でプロトコルスタックを持っている



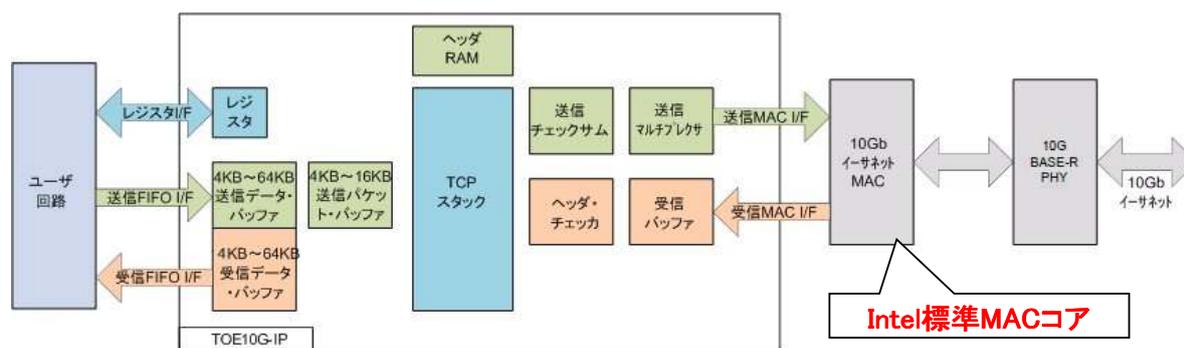
- TCP/IPの実装デメリット
  - TCPのプロトコル処理が複雑でCPUの負荷が大きい
  - パフォーマンスが上がらない(実効転送効率30%~40%)
  - ハイエンドの高価なCPUが必要となる



⇒ TOE10G-IPがこの悩みを解決します!

## TOE10G-IPコアの概要

- ・ 10GbE対応のTCP/IPオフロード・エンジン・コア
- ・ ユーザ回路とIntel製EMACコアの間に挿入
- ・ TCPの送信・受信の両方向の処理を完全HW化
- ・ Full Duplex(送受信同時)通信をサポート



TOE10G-IPコア・ブロック図

## TOE10G-IPコアの特長1

- ・ TCP/IP処理を完全ロジック・ハードウェア化
  - CPUなしでの組込みシステム実装が可能！
  - CPUシステムではCPU負荷がゼロ!
- ・ 送信/受信を高速転送
  - 約1200MByte/secの実転送パフォーマンス
- ・ 転送データの信頼性を確保
  - 送信時: ACK抜け/重複/タイムアウト等で自動リトライ
  - 受信時: シーケンス番号を評価し返信ACKを自動制御
- ・ データバッファ容量を選択可能
  - FPGAメモリ・リソースとパフォーマンスからユーザが選択



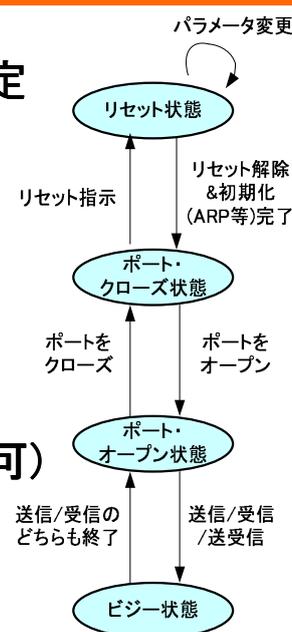
## TOE10G-IPコアの特長2

- ・ Intel純正MACコア(LL Ethernet 10G MAC)に対応
  - より低レイテンシ・低価格の弊社製10GEMACにも対応
- ・ 実機動作リファレンス・デザインを用意
  - Intel評価ボードで動作するプロジェクト
  - 購入前にパフォーマンス・信頼性の実機評価を検証可能
  - 製品のリファレンスはコア以外の全回路をソースで添付
  - サーバー・アプリ向けのマルチ・セッション・デザインあり
- ・ 10GBASE-Rに加え10GBASE-Tに対応
  - 低コストのCat6ケーブル・RJ45コネクタを適用可能



## TOE10G-IPコアの動作概要

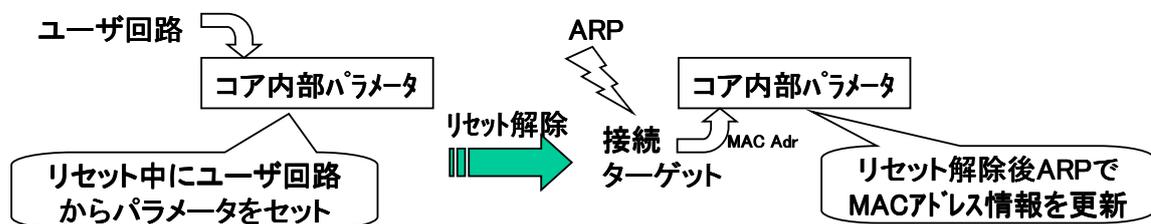
- ・ リセット状態でパラメータ(IP&MACアドレス等)を設定
- ・ リセット解除で初期化(ARP等)を実行
- ・ 初期化完了後ポートをオープン(アクティブ/パッシブ)
- ・ 送信・受信は独立して動作(同時送受信可)
- ・ パラメータ変更はリセット状態で実施  
(転送長/パケット長はビジー以外の状態で変更可)



コアの状態遷移図

## 初期化動作

- ・ パラメータの初期値設定
  - コアのリセット維持中にユーザ回路より設定
  - IPおよびMACアドレス・ポート番号を指定
  - 設定を完了するとリセットを解除
- ・ リセット解除後ARP実行
  - 接続ターゲットに対してARPを発行
  - 実行結果からターゲットMACアドレス情報を自動更新



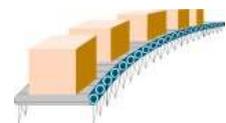
2021/2/19

Design Gateway

Page 9

## 高速送信

- ・ 送信パケットの生成
  - ユーザ回路は送信データをFIFO I/Fで書込み
  - 送信データをフレームサイズで分割
  - ヘッダと送信データを結合し10G MACへ出力
- ・ 自動再送機能
  - ターゲットからのACKを常時チェック
  - ACK抜け/重複/タイムアウト等の異常ACKを検出
  - 異常ACKの種類に応じて適切な再送データを用意



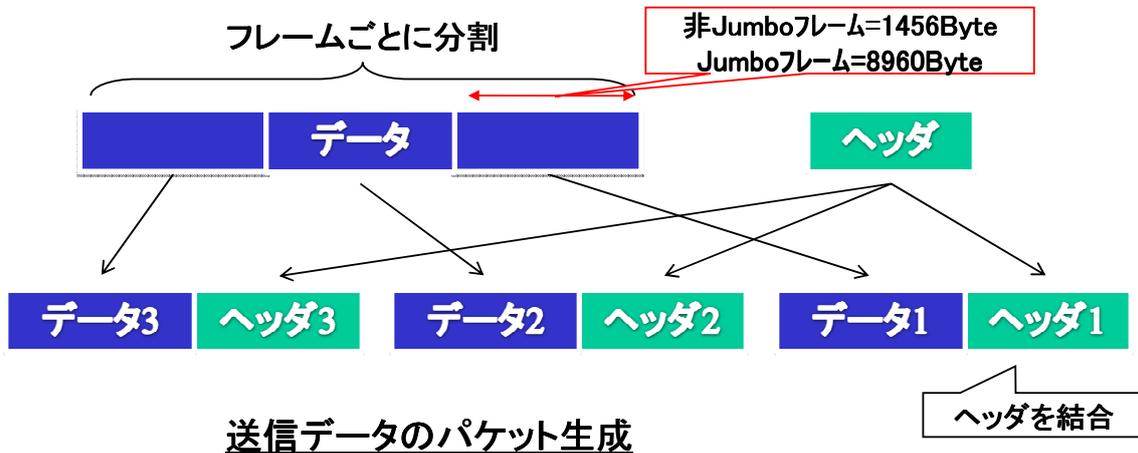
2021/2/19

Design Gateway

Page 10

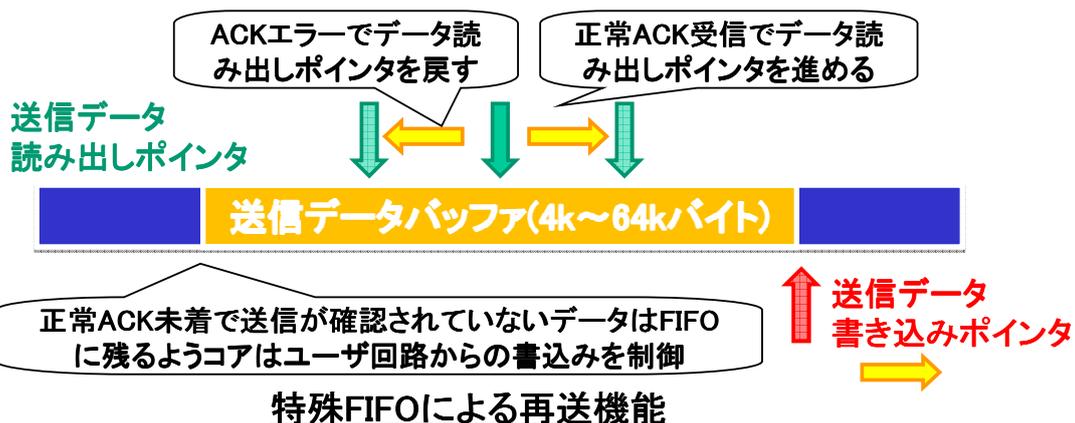
## 送信パケットの生成

- 送信データにヘッダを自動的に付加しEMACへ転送
  - データはコア内にてフレーム・サイズで分割
  - チェックサムやシーケンス番号などもコアが自動生成



## 自動再送機能

- 専用設計された特殊なFIFOで再送を実装
  - 正常ACKで読み出しポインタを進める
  - 異常ACKの場合コアは適切な位置にポインタを戻す
  - ポインタ制御・再送の実行をコアが自動制御



## 高速受信

- ・ 受信パケットのヘッダ・チェック 
  - 受信対象でない場合やチェックサム・エラーの場合は破棄
- ・ データ並び替え 
  - シーケンス番号の入れ替え発生時に並び替え
  - 受信済みデータの再送要求を防ぎ転送効率を維持
  - 並び替え処理できない場合は重複ACKで再送要求
- ・ 重複データの結合 
  - 受信データが前回受信パケットと重複する部分を検出
  - 重複箇所のみ破棄し連続データとして復元

## 受信パケットのヘッダ・チェック

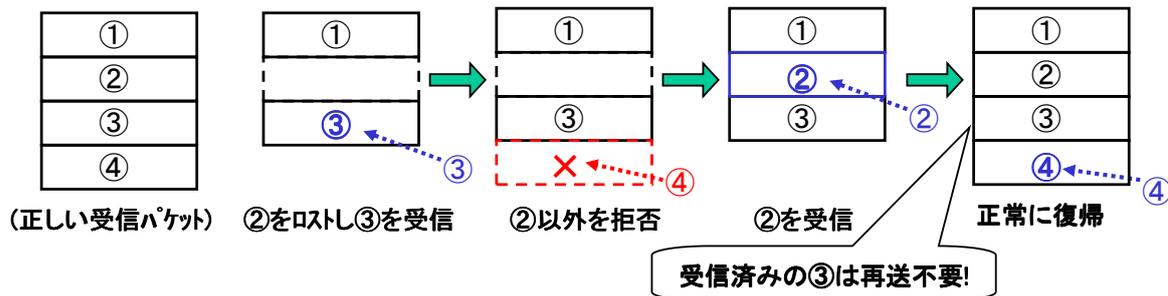
- ・ 受信ヘッダのチェック・サムが正しいことを確認
  - 更に以下の条件を満たすこともコアが確認

| Byteオフセット | プロトコル | データ内容         | 確認条件                                     |
|-----------|-------|---------------|------------------------------------------|
| 0-5       | ICMP  | あて先MACアドレス    | SML/SMHLレジスタで設定したコアのMACアドレスと一致           |
| 6-11      | ICMP  | 送信元MACアドレス    | ARPで検出した通信ターゲットのMACアドレスと一致               |
| 12-13     | ICMP  | タイプ           | 0x0800 (IPパケット)であること                     |
| 14        | IP    | バージョン/ヘッダ長    | 0x45 (IPv4, IPヘッダ長=20)であること              |
| 20        | IP    | フラグ/フラグメントOFS | 下位6ビットがゼロ(フラグメントの有無チェック)であること            |
| 23        | IP    | プロトコル番号       | 0x06(TCPパケット)であること                       |
| 26-29     | IP    | 送信元IPアドレス     | DIレジスタで設定した通信ターゲットのIPアドレスと一致             |
| 30-33     | IP    | あて先IPアドレス     | SIレジスタで設定したコアのIPアドレスと一致                  |
| 34-35     | TCP   | 送信元ポート番号      | DPNレジスタで設定/パッシブオープンで取得した通信ターゲットのポート番号と一致 |
| 36-37     | TCP   | あて先ポート番号      | SPNレジスタで設定したコアのポート番号と一致                  |
| 38-41     | TCP   | シーケンス番号       | 前回処理時のシーケンス番号との差分がコアで処理可能範囲内             |

### 受信パケット・ヘッダの確認条件

## データ並び替え

- シーケンス番号がスキップした(ロスト)場合に機能
  - ロスト状態を解消するパケット以外は受信しない
  - スキップ部のデータを回復するロスト解消パケットのみ受信
- データ並び替え
  - ロスト解消パケットからデータの連続性を完全に復元
  - 受信済みデータは再送要求せずパフォーマンスを維持



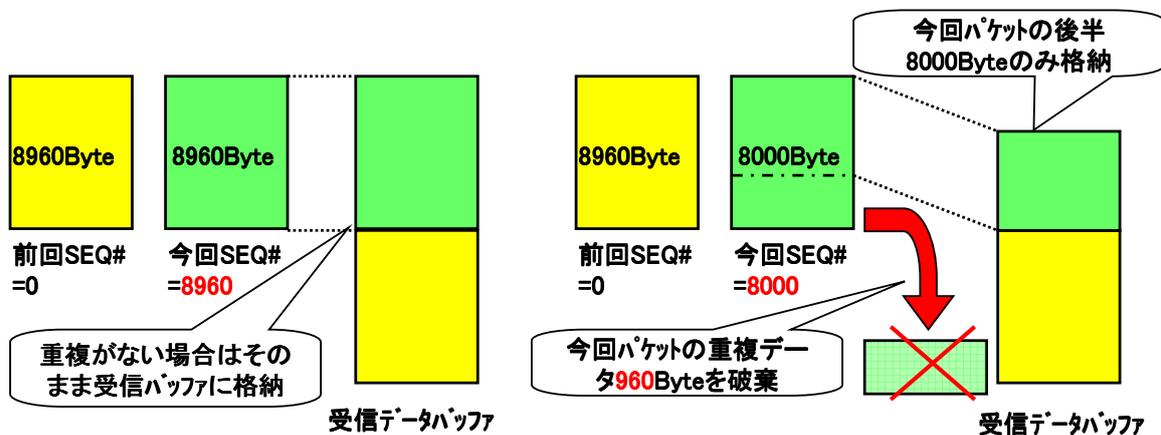
2021/2/19

Design Gateway

Page 15

## 重複データの結合

- データの重複を検出し自動的に補正
  - シーケンス番号から受信データの重複を検出
  - 重複箇所のみ破棄し連続データとして復元



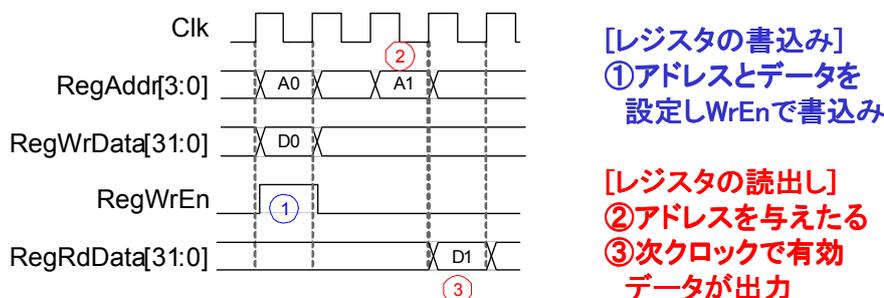
2021/2/19

Design Gateway

Page 16

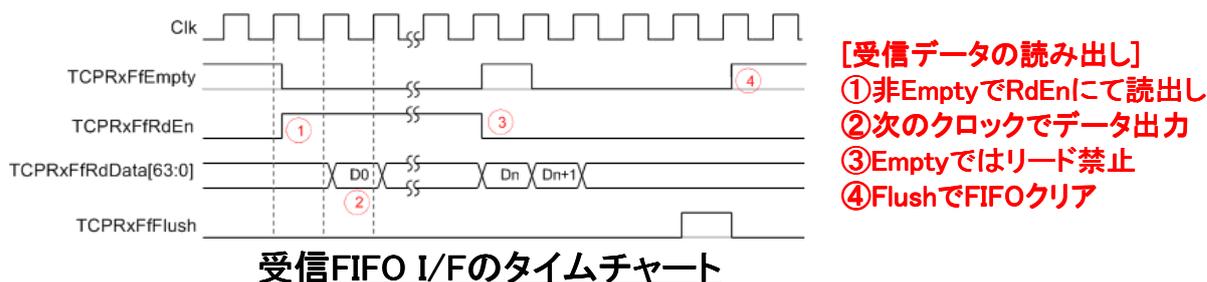
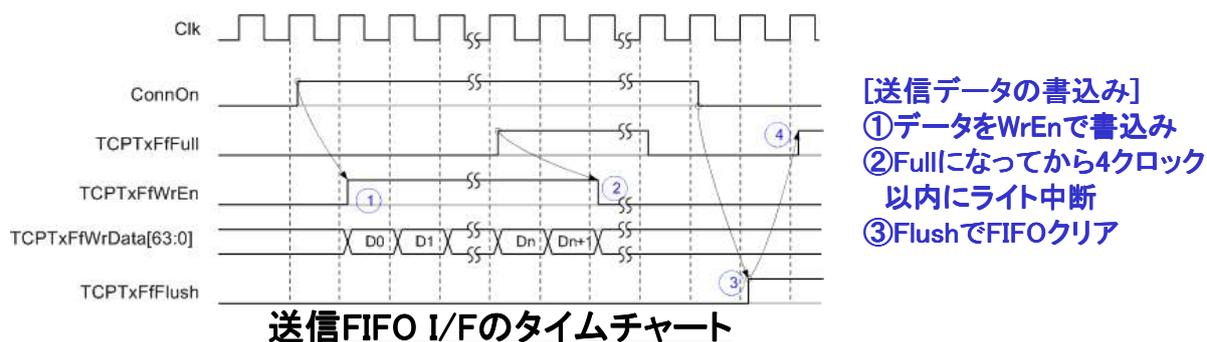
## ユーザ・インターフェース(制御)

- レジスタI/F、送信FIFO I/F、受信FIFO I/Fの3種類
  - レジスタI/Fは初期パラメータの設定、方向切り替え指示
  - 送信データ・受信データ用I/Fは標準的なFIFO I/F



レジスタI/Fのタイムチャート

## ユーザ・インターフェース(データ)



## データ・バッファ容量の設定

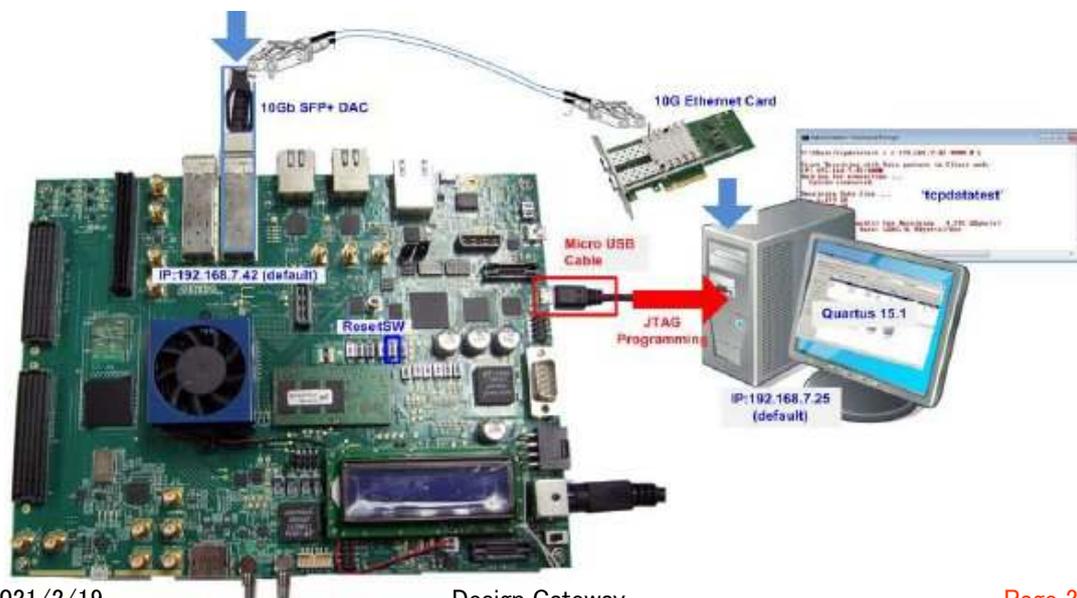
- ・ 3種類のデータ・バッファをパラメタライズで設定可能
  - ① 送信データ・バッファ: 4KByte~64KByte
  - ② 送信パケット・バッファ: 4KByte~16KByte
  - ③ 受信データ・バッファ: 4KByte~64KByte
- ・ リソースとパフォーマンスの最適点を調整できる

| ジェネリック名       | 設定範囲 | 説明                                                             |
|---------------|------|----------------------------------------------------------------|
| TxBufBitWidth | 9-13 | 送信データ・バッファ・サイズをアドレス・ビット幅で設定します。例えば9の場合4Kバイト、13の場合64Kバイトとなります。  |
| TxPacBitWidth | 9-11 | 送信パケット・バッファ・サイズをアドレス・ビット幅で設定します。例えば9の場合4Kバイト、11の場合16Kバイトとなります。 |
| RxBufBitWidth | 9-13 | 受信データ・バッファ・サイズをアドレス・ビット幅で設定します。例えば9の場合4Kバイト、13の場合64Kバイトとなります。  |

各データ・バッファはパラメタライズで設定できる

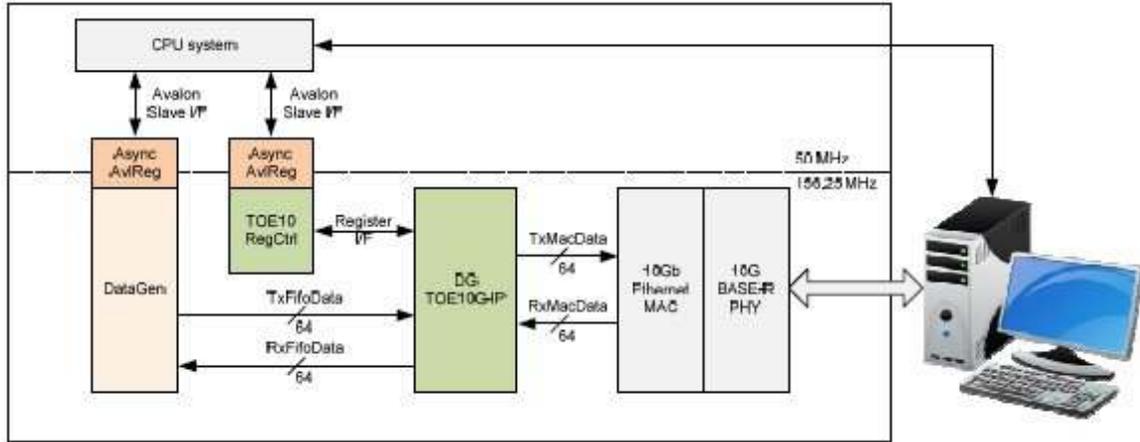
## 評価用SOFファイル

- ・ Arria10SoC開発ボードで動作するSOFファイル
  - 転送パフォーマンス測定・データベリファイ確認



# リファレンス・デザイン

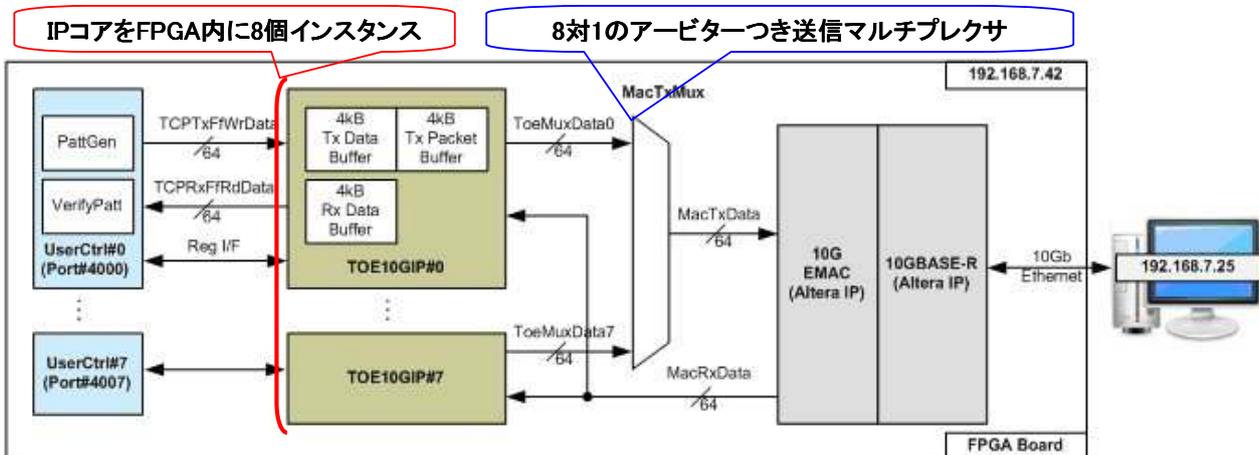
- ・ 実機動作するデザインプロジェクト
  - コア部以外の全回路をソースコードで製品に標準添付



リファレンス・デザイン・ブロック図

# マルチ・セッション対応

- ・ サーバー・アプリケーション参照用デザイン
  - コアを複数インスタンスでマルチ・セッションに対応



マルチセッション・デザイン(実機動作するデザインで提供)

## リファレンスと実機評価による開発

- ・ リファレンス+評価ボードによる確実な開発
  - まず最初に製品添付のリファレンスで実機動作を確認
  - そこからユーザ製品に向け少しずつ編集
  - 編集ごとに実機動作をStep by Stepで確認
  - 問題があれば1ステップ前に戻るだけで動く状態にすぐ復帰できる



**大きな後戻りがなく確実に短期間での製品開発が可能!**

## 10GBASE-Tに対応

- ・ 低価格CAT6ケーブル/RJ45コネクタ適用可能
  - 実パフォーマンスは10GBASE-Rとほぼ同等レベル

| 10GbE規格   | 送信          | 受信          | 送受信同時           |
|-----------|-------------|-------------|-----------------|
| 10GBASE-R | 1219MByte/s | 1109MByte/s | 800-1000MByte/s |
| 10GBASE-T | 1205MByte/s | 1103MByte/s | 800-1000MByte/s |

### TOE10G-IPコアの実パフォーマンス測定結果

測定条件: PC-Arria10GXボード間のリファレンス・デザイン測定結果

パケットサイズ: 8960byte (Jumbo Frame)

10GBASE-Tの測定ではA10GXボードのSFP+にRJ45変換アダプタ使用(10GTek製 ASF-10G-T)

PC側スペック: CPU Intel i5-8500@3.00GHz, メモリDDR4 16GB, Windows10Pro, NICカード: X550-T1(Intel)

※ 送受信同時(Full Duplex)のパフォーマンスは800-1000MB/sの範囲で変化する

# 消費リソース

## ・ FPGAのコア単体消費リソース

- 送信データ・バッファ=64KByte、送信パケット・バッファ=16KByte、受信データ・バッファ=64KByteの最大設定時



| Family        | Example Device     | Fmax (MHz) | ALMs <sup>1</sup> | Registers <sup>1</sup> | Pin | Block Memory bit <sup>2</sup> | Design Tools  |
|---------------|--------------------|------------|-------------------|------------------------|-----|-------------------------------|---------------|
| Arria 10 SX   | 10AS066N3F40E2SGE2 | 156.25     | 2,566             | 3,931                  | -   | 1,179,648                     | QuartusII16.0 |
| Arria 10 GX   | 10AX115S2F45I2SG   | 156.25     | 2,453             | 3,491                  | -   | 1,179,648                     | QuartusII16.0 |
| Cyclone 10 GX | 10CX220Y7F80E5G    | 156.25     | 2,247             | 3,446                  | -   | 1,179,648                     | QuartusII18.0 |
| Stratix 10 GX | 1SG280HU2F50E2VGS1 | 156.25     | 2,928             | 3,523                  | -   | 1,179,648                     | QuartusII18.0 |

### TOE10G-IPコア単体コンパイル結果

送受信ともバッファを最大に設定した場合です。  
バッファ容量を削減すればMemory消費リソースを節約できます。

# 送信パフォーマンス

```

/cygdrive/d/altera/15.1
+++ TOE10G-IP Send Mode +++
Enter transfer size : 1 - 0xFFFFFFFF => 0xffffffff Input parameter to test
Enter packet size (aligned 64-bit) : 8 - 8960 => 8960 send data in Server mode
Wait connection from PC...
Run test application on PC by following command
tcpdatatest c r 192.168.7.42 4000 0 1
Start data sending
...
Connection closed
Total user transfer size = 4294967295 byte
Total = 4294[MB] , Time = 3569[ms] , Transfer speed = 1203[MB/s] Send Performance

--- TOE10GIP menu ---
[0] : Display current parameter
[1] : Reset TOE10G-IP to change parameter or mode
[2] : Send Data Test (TOE10G-IP -> PC)
[3] : Receive Data Test (PC -> TOE10G-IP)

```

4Gバイト・データをジャンボフレーム  
(8960バイト)でFPGA->PCへ転送

送信パフォーマンス  
1203MByte/sec!

### Arria10SoCボードでの送信(FPGA->PC)評価結果

# 受信パフォーマンス

```

/cygdrive/d/altera/15.1

+++ TOE10G-IP Receive Mode +++
Enter transfer size : 1 - 0xFFFFFFFF => 0xffffffff
Enable data verification : [0] Disable [1] Enable => [0] Disable data verification
Wait connection from PC...
Run test application on PC by following command
tcpdatatest c t 192.168.7.42 4000 4294967295 0
Connection opened
...
Connection closed
Wait UserLogic complete
Receive data completed

Total user transfer size = 4294967295 byte
Total = 4294[MB] , Time = 3662[ms] , Transfer speed = 1172[MB/s]
Receive Performance without verification

--- TOE10GIP menu ---
[0] : Display current parameter
[1] : Reset TOE10G-IP to change parameter or mode
[2] : Send Data Test (TOE10G-IP -> PC)
[3] : Receive Data Test (PC -> TOE10G-IP)
    
```

4Gバイト・データをジャンボフレームでPC->FPGAへ転送

受信パフォーマンス  
1172MByte/sec!

## Arria10SoCボードでの受信(PC->FPGA)評価結果

# TOE10G-IPのアプリケーション

- ・ FAなどにおけるデータ転送
  - 医療系の画像処理装置の採用例が多い
  - 欠落が許されないセンサーデータなどをPCに転送
- ・ NAS, iSCSIなどのTCPを用いたストレージ
  - TCPの負荷を大幅に軽減し、転送速度UP
- ・ 監視カメラデータの転送
  - TCPの負荷が軽くなり、コストダウン、低消費電力化が可能

