

Lattice CrossLink ユーザーズガイド



- 本資料は、Lattice社CrossLink使用時の注意事項、注意事項をまとめたものです。実際の動作等詳細、最終の確認は、別途データシート、テクニカルノートを参照頂けるようお願い致します。
- Lattice社データシート及びテクニカルノートと本資料との間に差異があった場合には、Lattice社データシートを正としてお取り扱い下さい。



	Page
<u>1. 電源ピンの処理</u>	
1-1. 電源ピン説明	5
1-2. 電源ピン注意事項	6
1-3. 電源シーケンス	7
<u>2. I/Oピンの処理</u>	
2-1. I/Oバンクの構成	9
2-2. サポートする入出力I/Fについて	10
2-3. ドライブ能力について	11
2-4. 未使用ピンの処理について	12
2-5. 内部プルアップ抵抗値について	13
2-6. クロック入力専用ピンについて	15
2-7. 差動信号のピンアサイン	16
<u>3. 高速I/Oインターフェース関連</u>	
3-1. Clock Networkについて	18
3-2. GDDRモジュールについて	19
3-3. GDDRサンプル(GDDR4 ECLK.Centered)	20
3-4. GDDRサンプル(GDDR71 ECLK)	24
3-5. Hard D-PHYモジュールについて	28
<u>4. コンフィグレーション関連</u>	
4-1. コンフィグ関連ピン説明	31
4-2. コンフィグレーションフローチャート	32
4-3. コンフィグレーションデータ容量	33
4-4. コンフィグレーション回路	34
4-5. Slave SPI Configuration	36
<u>5. 内部波形観測ツール“Reveal”使用時の注意点</u>	
5-1. Reveal用I/Oの確保について	40
Revision History	41

電源ピンの処理

1-1 電源ピン説明

ピン名	電圧 (Min)	電圧 (Max)	備考
VCC	1.14V	1.26V	コア供給電源
VCCPLL	1.14V	1.26V	PLL供給電源
VCCAUX	2.375V	2.625V	Bank 1, 2, 及びNVCM用補助電源
VCCIO	1.14V	3.465V	Bank 0, 1, 2用I/O電源
D-PHY電源			
VCC_DPHY	1.14V	1.26V	D-PHY供給電源
VCCA_DPHY	1.14V	1.26V	D-PHYアナログ電源
VCCPLL_DPHY	1.14V	1.26V	D-PHY用PLL電源
VCCMU_DPHY	1.14V	1.26V	VCCA_DPHY, VCCPLL_DPHY共通電源。WLCSP36パッケージにのみ存在。

- 同じ電圧の電源は共通電源を使用して下さい。（例：電源電圧が同じVCCIO、VCCとVCCPLLとD-PHY電源。）
- VCC、VCCPLL、VCCA_DPHY、VCCPLL_DPHY、VCCMU_DPHYは共通の1.2V電源を使用しますが、フェライトビーズ等で分離するようにして下さい。
- VCCIO0 (Bank0電源電圧) はコンフィグレーションI/F電圧と合わせるようにして下さい。（例：SPI FlashやSlave SPI I/Fの信号レベルが3.3Vであれば3.3Vを供給。）
- PLLやD-PHYを使用しない場合でも、全ての電源に規定の電圧を供給して下さい。

■ VCCIOに関して

未使用バンクについても電源電圧を供給するようにして下さい。
また、各スタンダードに応じて以下のように個別に入力電圧範囲が決まっていますので、この電源電圧範囲を守って下さい。

Standard	Vccio		
	Min	Typ	Max
LVC MOS33/LVTTL33	3.135	3.30	3.465
LVC MOS25	2.375	2.50	2.625
LVC MOS18	1.710	1.80	1.890
LVC MOS12 (Output only)	1.140	1.20	1.260
subLDVS (Input only)	2.375	2.50	2.625
SLVS (Input only) ²	2.375	2.50	2.625
LVDS	2.375	2.50	2.625
MIPI (Input only)	1.140	1.20	1.260

VCCIOはVCC及びVCCAUXより先、または同時に立ち上げて下さい。

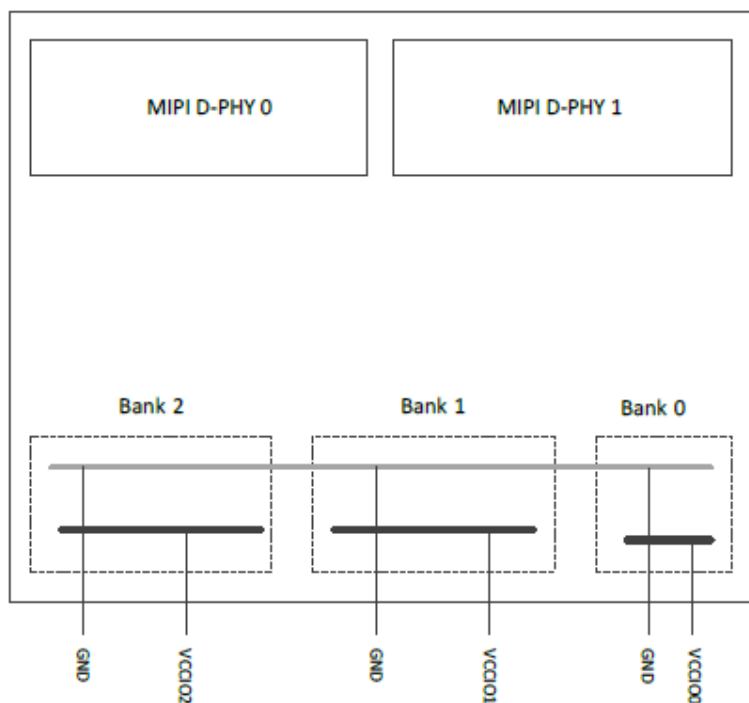
VCCAUXを除く全ての電源は以下のランプレートを守り、単調増加で立ち上げて下さい。

Symbol	Parameter	Min	Max	Unit
t_{RAMP}	Power supply ramp rates for all power supplies except $V_{CCAUX25VPP}$	0.6	10	V/ms

Power On Reset (以下POR) はVCC、VCCIO0、VCCAUXが以下の電圧まで立ち上がると有効になり、リリース後にコンフィグレーションが開始されます。その他の電源はPORには関係していません。

Symbol	Parameter	Min	Typ	Max	Unit	
V_{PORUP}	Power-On-Reset ramp up trip point (Monitoring V_{CC} , V_{CCIO0} , and V_{CCAUX})	V_{CC}	0.62	0.68	0.93	V
		V_{CCIO0}^2	0.87	1.08	1.50	V
		V_{CCAUX}	0.90	—	1.53	V
V_{PORDN}	Power-On-Reset ramp down trip point (Monitoring V_{CC} , V_{CCIO0} , and V_{CCAUX})	V_{CC}	—	—	0.79	V
		V_{CCIO0}^2	—	—	1.50	V
		V_{CCAUX}	—	—	1.53	V

I/Oピンの処理



- CrossLinkはプログラマブルI/O Bankとして、Bank0~2を持っており、各Bank毎に電源（VCCIO）が用意されています。D-PHY専用バンクは0~1の2バンク構成になっています。
※WLCSP36はBank1、及びD-PHY 0を持っていません。

- 各バンクで使用するI/Oスタンダードに応じてバンク電源（VCCIO）に適切な電圧を入力して下さい（1-2項参照）。

- 差動信号入出力はBank1, Bank2でのみ可能です。

- 各I/Oで内部プルアップが使用可能です。ツール設定で以下のプルアップの有効、無効が設定できます。

3P3K (3.3 k Ω)

6P8K (6.8 k Ω)

10K (10 k Ω)

UP (100 k Ω)

NONE

- ※ 3P3K、6P8K、10KはBank 0でのみ選択できます。また、UP (100K Ω)は大よそのプルアップ抵抗値です。詳細は2-5項を参照して下さい。

- **CrossLinkのI/Oは活栓挿抜に対応していません。** 電源OFF時に外部から信号を入力することはできません。

- CrossLinkのI/Oはコンフィグレーション中はHi-zで内部プルアップが有効となっています。

Description (Types of I/O Buffers)	Top Side D-PHY Banks	Bottom Side Bank 0	Bottom Side Bank 1 and Bank 2
Single-ended Output Standards	—	LVTTTL LVCMOS33 LVCMOS25 LVCMOS18 LVCMOS12	LVTTTL LVCMOS33 LVCMOS25 LVCMOS18 LVCMOS12
Differential Output Standards	MIPI D-PHY SLVS	—	LVDS LVTTTL33D LVCMOS33D LVCMOS25D
Single-ended Input Standards	—	LVTTTL LVCMOS33 LVCMOS25 LVCMOS18	LVTTTL LVCMOS33 LVCMOS25 LVCMOS18
Differential Input Standards	MIPI D-PHY SLVS	—	LVDS SUBLVDS SLVS LVCMOS25D
On Die Termination	Yes	No	Yes (only for LVDS)

CrossLinkは左の表の通り各Bank毎にサポート可能なI/Fが異なります。

また、**CrossLinkはMulti Voltage Inputに対応していません。**各入力スタンダードに応じたVCCIOを供給する必要があります。

例えば、3.3V LVCMOSインターフェースを取る際は該当バンクのVCCIOに3.3Vを供給する必要があり、またそのバンクに属するI/Oに3.3V以外（2.5V、1.8V等）の信号を入力することはできません。

Standard	V _{CCIO}		
	Min	Typ	Max
LVCMOS33/LVTTTL33	3.135	3.30	3.465
LVCMOS25	2.375	2.50	2.625
LVCMOS18	1.710	1.80	1.890
LVCMOS12 (Output only)	1.140	1.20	1.260
subLDVS (Input only)	2.375	2.50	2.625
SLVS (Input only) ²	2.375	2.50	2.625
LVDS	2.375	2.50	2.625
MIPI (Input only)	1.140	1.20	1.260

V _{CCIO}	Dependent IO Type	Drive Strength (mA)
3.3	LVC MOS33	8
2.5	LVC MOS25	6
1.8	LVC MOS18	4
1.2	LVC MOS12	2

CrossLinkのドライブ電流は

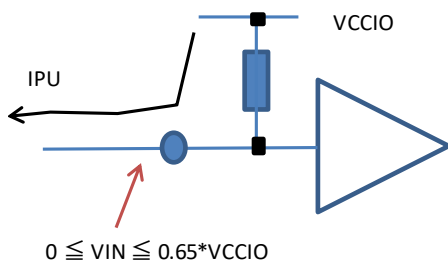
LVC MOS33 8 mA
 LVC MOS25 6 mA
 LVC MOS18 4 mA
 LVC MOS12 2 mA

となっており、全て固定値でツールでの変更はできません。

- CrossLinkの未使用ピンは全てオープンで構いません。
- Bank0～3の未使用ピンは出力バッファがHi-z出力で、かつ内部プルアップが有効になっており、入力バッファはこの内部プルアップによって保護されています。I/Oパッドにはこの内部プルアップによるHighが見える状態になります。
- D-PHY Bankの未使用ピンは出力Hi-zで内部プルアップ無しですが、外部処理はオープンで問題ありません。

2-5 内部プルアップ抵抗値について

CrossLinkの内部プルアップ（UP（100K Ω ））の詳細な抵抗値はプルアップ抵抗に流れるリーク電流と、Vccioの推奨電圧範囲から求められます。



Symbol	Parameter	Condition	Min	Typ	Max	Unit
I_{pu}^4	Internal Pull-Up Current	$V_{CCIO} = 1.2 \text{ V between } 0 \leq V_{IN} \leq 0.65 * V_{CCIO}$	-2.7	—	-31	μA
		$V_{CCIO} = 1.8 \text{ V between } 0 \leq V_{IN} \leq 0.65 * V_{CCIO}$	-3	—	-31	μA
		$V_{CCIO} = 2.5 \text{ V between } 0 \leq V_{IN} \leq 0.65 * V_{CCIO}$	-8	—	-72	μA
		$V_{CCIO} = 3.3 \text{ V between } 0 \leq V_{IN} \leq 0.65 * V_{CCIO}$	-11	—	-128	μA

Standard	V_{CCIO}		
	Min	Typ	Max
LVC MOS33/LVTTL33	3.135	3.30	3.465
LVC MOS25	2.375	2.50	2.625
LVC MOS18	1.710	1.80	1.890
LVC MOS12	1.140	1.20	1.260

I_{pu} が最大になるのは $V_{IN} = 0\text{V}$ として、オームの法則より以下のプルアップ抵抗値が求まります。

	VCCIO	IPU	$R_{pu_eff} [k\Omega]$
LVC MOS33	3.465	128	27.070
LVC MOS33	3.135	128	24.492
LVC MOS25	2.625	72	36.458
LVC MOS25	2.375	72	32.986
LVC MOS18	1.89	31	60.968
LVC MOS18	1.71	31	55.161
LVC MOS12	1.26	31	40.645161
LVC MOS12	1.14	31	36.774194

I_{pu} が最小になるのは $V_{IN} = 0.65 * V_{CCIO}$ として、オームの法則より以下のプルアップ抵抗値が求まります。

	VCCIO	V_{Δ}	IPU	$R_{pu_eff} [k\Omega]$
LVC MOS33	3.465	1.21275	11	110.250
LVC MOS33	3.135	1.09725	11	99.750
LVC MOS25	2.625	0.91875	8	114.844
LVC MOS25	2.375	0.83125	8	103.906
LVC MOS18	1.89	0.6615	3	220.500
LVC MOS18	1.71	0.5985	3	199.500
LVC MOS12	1.26	0.441	3	147.000
LVC MOS12	1.14	0.399	3	133.000

※ $V_{\Delta} = 0.35 * V_{CCIO}$

以上から、内部プルアップ抵抗値は次のように算出されます。

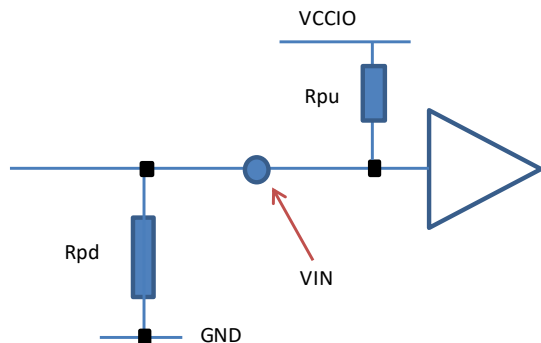
24.492K Ω ~ 110.250K Ω @ LVC MOS33

32.986K Ω ~ 114.844K Ω @ LVC MOS25

55.161K Ω ~ 220.500K Ω @ LVC MOS18

36.774K Ω ~ 147.000K Ω @ LVC MOS12

コンフィグレーション中など、内部プルアップON、出力Hi-zの時は内部プルアップによるHighが現れますが、これを外部プルダウンでLowに固定するには、内部プルアップ抵抗値が最小時に外部プルダウンでピンの電位をVIL(Max)以下となるようにプルダウン抵抗値を決定します。



抵抗分圧式 $V_{IN} = V_{CCIO} * R_{pd} / (R_{pu} + R_{pd})$ より、内部プルアップON時にLowレベル確定するには、

$$V_{CCIO} * R_{pd} / (R_{pu_min} + R_{pd}) < V_{ILmax}$$

を満たせばよく、変換すると

$$R_{pd} < R_{pu_min} * V_{ILmax} / (V_{CCIO} - V_{ILmax})$$

を満たせば良いことになり、以下のように算出されます。

8.373KΩ @ LVCMOS33
13.827KΩ @ LVCMOS25
29.702KΩ @ LVCMOS18
19.801KΩ @ LVCMOS12

※全てのケースにおいて、2.2K程度のプルダウン抵抗を用意すれば問題ありません。

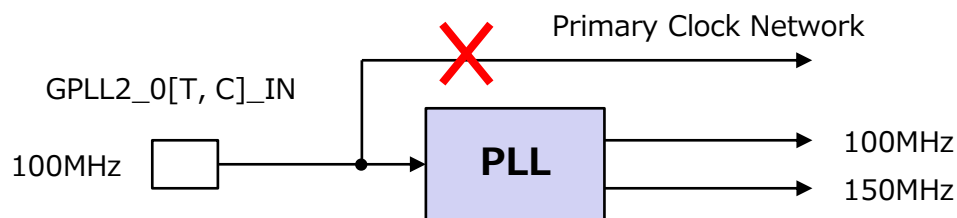
※前ページより、内部プルアップ抵抗値(Min)は以下の通り
24.492KΩ @ LVCMOS33
32.986KΩ @ LVCMOS25
55.161KΩ @ LVCMOS18
36.774KΩ @ LVCMOS12

※VIL (Max) は以下の通り

Input/Output Standard	V _{IL}	
	Min (V)	Max (V)
LVCMOS33/ LVTT133	-0.3	0.8
LVCMOS25	-0.3	0.7
LVCMOS18	-0.3	0.35 V _{CCIO}
LVCMOS12	-0.3	0.35 V _{CCIO}

ピン名	I/O方向	説明
GPLL2_0[T, C]_IN	I	PLLに最短パスで接続されるクロック入力専用ピンです。Primary Clock Networkに直接配線することはできません。
GR_PCLK[Bank]0	I	直接Primary Clock Networkには接続されず一般配線領域に配線されますが、他の汎用I/Oよりも短いパスでPrimary Clock Networkに配線されます。
PCLK[T/C][Bank]_[num]	I/O	直接Primary Clock Networkに接続される専用クロックピンです。
MIPI_CLK[T/C][Bank]_0	I/O	MIPI D-PHYハードマクロ内のPLLのリファレンスクロック入力ピンです。外部からD-PHYのリファレンスクロックを入力する場合に使用します。

- ・ CrossLinkには上記クロック入力専用ピンが用意されています。
- ・ 差動クロックをアサインする場合、T側に+、C側に- をアサインして下さい。
シングルエンド入力の場合は必ずT側にアサインして下さい。
- ・ GPLL2_0[T, C]_INからのクロック入力は直接Primary Clock Networkに配線されません。そのため、以下のようなPLL入力を分岐して他のPrimary Clock Networkを使用するFFに使用することはできません（設計ツールDiamond上でエラーとなります。）



入力クロックの分岐はできないため、同じ周波数のクロックを使用する場合は同じ周波数のクロック出力を用意します。

- ・ 通常、PLLへの入力クロックはPCLKにアサインすれば問題ありません。

- Bank1, Bank2の汎用I/Oにアサインする場合、差動信号はPin Function名の末尾がAまたはCのピンに+側、末尾がBまたはDのピンに-側をアサインして下さい。また、A/B、C/Dを差動ペアとしてアサインして下さい。

(例) PB29A : 差動信号+ をアサイン
PB29B : 差動信号- をアサイン

PB6C : 差動信号+ をアサイン
PB6D : 差動信号- をアサイン

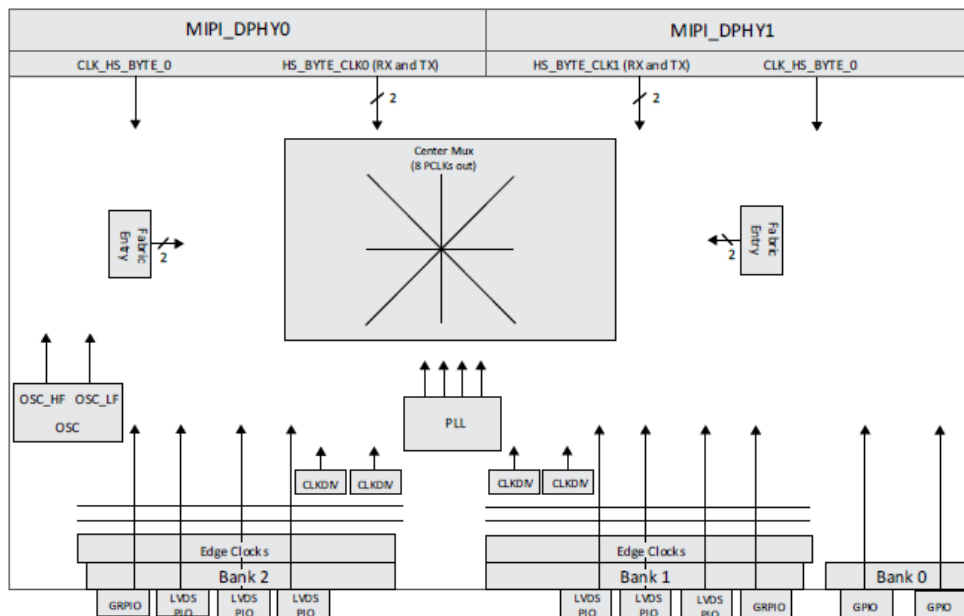
B、Dに+側をアサインしたり、A/D、B/Cというペアにすることはできません。

- 高速ソースシンクロナスI/Fの送受信で使用するGDDRモジュール（GDDR2、GDDR4、GDDR8、GDDR71、GDDR141）はA/Bペア、C/Dのどちらでも使用可能です。
- Bank0では差動信号のアサインや、GDDRモジュールの使用はできません。
- D-PHYハードマクロへのピンアサインは、P側に差動信号+側、N側に差動信号-側をアサインして下さい。

MIPI D-PHY		
DPHY[num]_CK[P/N]	I/O	MIPI D-PHY Clock [num] = D-PHY 0 or 1, P = Positive, N = Negative.
DPHY[num]_D[P/N][lane]	I/O	MIPI D-PHY Data [num] = D-PHY 0 or 1, P = Positive, N = Negative, Lane = data lane in the D-PHY block 0, 1, 2 or 3.

高速I/Oインターフェース関連

3-1 Clock Networkについて



- CrossLinkには8本のPrimary Clock Networkが用意されています。このクロック専用配線は一般配線より低スキューでフリップフロップを駆動します。
- Primary Clock NetworkにはI/Oからの入カクロック、内部生成クロック、PLL出カクロック、内部オシレータ出力、CLKDIV出力、D-PHYハードマクロからの出カクロック等、全てのクロックを接続することが可能です。
- Bank1、Bank2のI/O近傍には各バンクに2本ずつEdge Clock (ECLK) が用意されています。このECLKは専用のECLKSYNCBマクロによってドライブされ、高速ソースシンクロナスI/F入出力用のGDDR_X2、GDDR_X4、GDDR_X8、GDDR71、GDDR141のDDRレジスタを最も低スキューで駆動します。ユーザーが意識してECLK、及びCLKSYNCBをアサインする必要は無く、Clarity Designerで該当のGDDRモジュールを生成すれば自動でアサインされます。
- Bank1とBank2を跨いだソースシンクロナスI/Fのアサインは、Bank1とBank2のECLKが共用されることで実現されます。これもユーザーが意識してアサインする必要はありません。

GDDRモジュールとはI/Oセルに用意されたDDRレジスタと後段のギアロジックで構成されるシリアル-パラレル、及びパラレル-シリアル変換回路です。CrossLinkには以下のGDDRモジュールが用意されています。

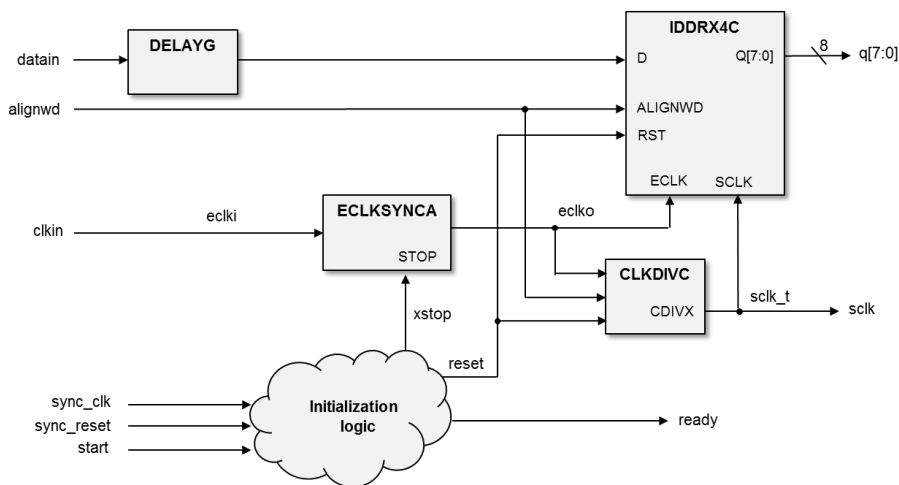
Rx Module	説明
GDDR1_RX.SCLK.Aligned	1:2シリアル-パラレル変換モジュールです。クロックエッジとデータエッジを同位相で入力します。
GDDR1_RX.SCLK.Centered	1:2シリアル-パラレル変換モジュールです。クロックエッジはデータのセンター位置で入力します。
GDDR2_RX.ECLK.Aligned	1:4シリアル-パラレル変換モジュールです。クロックエッジとデータエッジを同位相で入力します。
GDDR2_RX.ECLK.Centered	1:4シリアル-パラレル変換モジュールです。クロックエッジはデータのセンター位置で入力します。
GDDR4_RX.ECLK.Aligned	1:8シリアル-パラレル変換モジュールです。クロックエッジとデータエッジを同位相で入力します。
GDDR4_RX.ECLK.Centered	1:8シリアル-パラレル変換モジュールです。クロックエッジはデータのセンター位置で入力します。
GDDR8_RX.ECLK.Aligned	1:16シリアル-パラレル変換モジュールです。クロックエッジとデータエッジを同位相で入力します。
GDDR8_RX.ECLK.Centered	1:16シリアル-パラレル変換モジュールです。クロックエッジはデータのセンター位置で入力します。
GDDR71_RX.ECLK	1:7シリアル-パラレル変換モジュールです。FPD-Link, OpenLDI等、データがクロックの7通倍レートで入力されるI/Fに使用します。
GDDR141_RX.ECLK	1:14シリアル-パラレル変換モジュールです。FPD-Link, OpenLDI等、データがクロックの7通倍レートで入力されるI/Fに使用します。
Tx Module	
GDDR1_TX.SCLK.Aligned	2:1パラレル-シリアル変換モジュールです。クロックエッジとデータエッジを同位相で出力されます。
GDDR1_TX.SCLK.Centered	2:1パラレル-シリアル変換モジュールです。クロックエッジはデータのセンター位置で出力されます。
GDDR2_TX.ECLK.Aligned	4:1パラレル-シリアル変換モジュールです。クロックエッジとデータエッジを同位相で出力されます。
GDDR2_TX.ECLK.Centered	4:1パラレル-シリアル変換モジュールです。クロックエッジはデータのセンター位置で出力されます。
GDDR4_TX.ECLK.Aligned	8:1パラレル-シリアル変換モジュールです。クロックエッジとデータエッジを同位相で出力されます。
GDDR4_TX.ECLK.Centered	8:1パラレル-シリアル変換モジュールです。クロックエッジはデータのセンター位置で出力されます。
GDDR8_TX.ECLK.Aligned	8:1パラレル-シリアル変換モジュールです。クロックエッジとデータエッジを同位相で出力されます。
GDDR8_TX.ECLK.Centered	8:1パラレル-シリアル変換モジュールです。クロックエッジはデータのセンター位置で出力されます。
GDDR71_TX.ECLK	7:1パラレル-シリアル変換モジュールです。FPD Link, OpenLDI等、データがクロックの7通倍レートで入力されるI/Fに使用します。
GDDR141_TX.ECLK	14:1パラレル-シリアル変換モジュールです。FPD Link, OpenLDI等、データがクロックの7通倍レートで出力されるI/Fに使用します。

GDDRモジュールの例としてGDDR4 ECLK.Centered (Rx 1:8変換、Tx 8:1変換、クロックはデータのセンター位置) の機能説明を以下に記載します。

【ポート仕様】

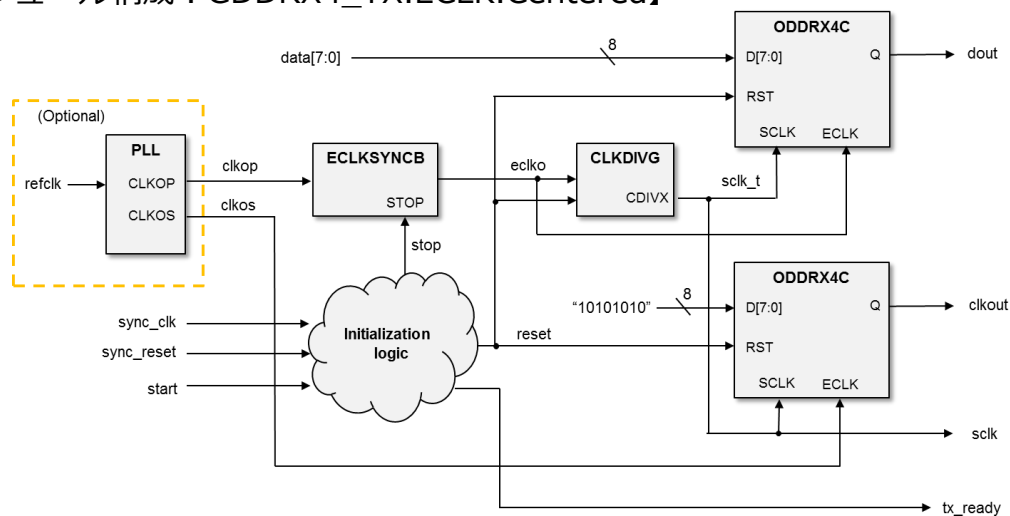
ポート名	I/O方向	説明	コメント
GDDR4_RX.ECLK.Centered			
clkln	Input	GDDR4 Rxソースシンクロナスクロック入力。	
sync_reset	Input	GDDRインターフェースイニシャライズロジックリセット入力 (Active High)。	
datain	Input	GDDR4 Rxシリアルデータ入力。	
alignwd	Input	パラレル出力データアライメント用入力 (Active High)。Activeにすると、シリアルデータの取り込みタイミングを1bit後ろにシフトします。	パルス幅はECLK x2以上確保する必要があります。
sync_clk	Input	GDDRインターフェースイニシャライズロジック用クロック。ECLKの1/2以下の周波数にする必要があります。	クロックソースはECLKと非同期でも問題ありません。sclkはsync_clkによって動作するイニシャライズシーケンス完了後に出力されるので使用できません。
start	Input	GDDRインターフェースイニシャライズロジックのトリガ信号 (Active High)。	イニシャライズシーケンスを開始する際にLow -> Highとトグルして下さい。
q	Output	GDDR4 Rxパラレルデータ出力。	
sclk	Output	GDDR4 Rxパラレルデータ出力に同期したシステムクロック。	
ready	Output	GDDR4 Rxイニシャライズ完了通知信号。	
GDDR4_TX.ECLK.Centered			
refclk	Input	GDDR4 Txモジュール内PLLクロック入力。	PLLをモジュール内に組み込む場合のみ存在。
clkop	Input	PLL clkop出カクロック入力ポート。	PLLをモジュール内に組み込まない場合のみ存在。出力データレート1/2の周波数にして下さい。
clkos	Input	PLL clkos出カクロック入力ポート。	PLLをモジュール内に組み込まない場合のみ存在。clkopを90° shiftしたクロックにして下さい。
sync_reset	Input	非同期リセット入力 (Active High)。	
tristate	Input	出カトリス状態制御信号 (Active High)。	
data	Input	GDDR4 Txパラレルデータ入力。	
sync_clk	Input	GDDRインターフェースイニシャライズロジック用クロック。ECLKの1/2以下の周波数にする必要があります。	クロックソースはECLKと非同期でも問題ありません。sclkはsync_clkによって動作するイニシャライズシーケンス完了後に出力されるので使用できません。
start	Input	GDDRインターフェースイニシャライズロジックのトリガ信号 (Active High)。	イニシャライズシーケンスを開始する際にLow -> Highとトグルして下さい。
dout	Output	GDDR4 Txシリアルデータ出力。	
clkout	Output	GDDR4 Txソースシンクロナスクロック出力。	
sclk	Output	GDDR4 Txパラレルデータ入力に同期したシステムクロック。	
ready	Output	GDDR4 Txイニシャライズ完了通知信号。	

【モジュール構成：GDDR4_RX.ECLK.Centered】



※ GDDR4_RXモジュールからの出力データは sclk に同期して出力されます。そのため、後段回路では必ず GDDR4_RXモジュールから出力される sclk でデータをラッチして下さい。

【モジュール構成：GDDR4_TX.ECLK.Centered】



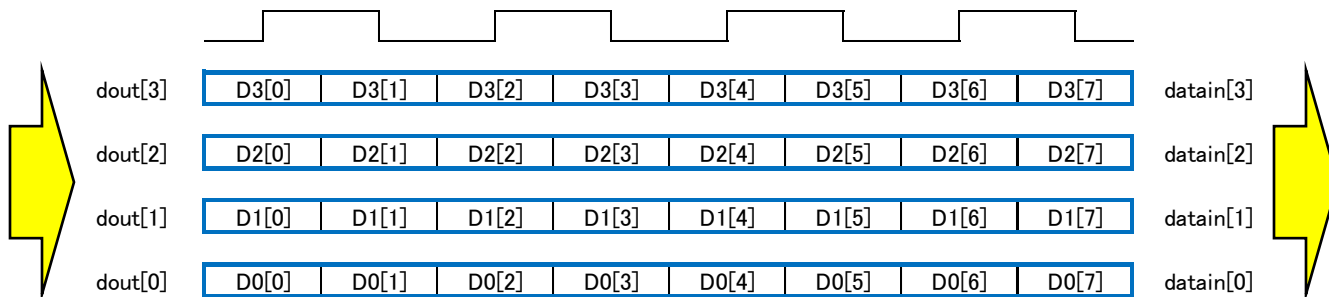
※ GDDR4_TXモジュール内では入力データは sclk でラッチされます。そのため、前段回路では必ず GDDR4_TXモジュールから出力される sclk に同期させるように設計して下さい。

【GDDR4入出力データ並び (By Lane)】

Tx

D3[7]	=	data[31]
D3[6]	=	data[30]
D3[5]	=	data[29]
D3[4]	=	data[28]
D3[3]	=	data[27]
D3[2]	=	data[26]
D3[1]	=	data[25]
D3[0]	=	data[24]
D2[7]	=	data[23]
D2[6]	=	data[22]
D2[5]	=	data[21]
D2[4]	=	data[20]
D2[3]	=	data[19]
D2[2]	=	data[18]
D2[1]	=	data[17]
D2[0]	=	data[16]
D1[7]	=	data[15]
D1[6]	=	data[14]
D1[5]	=	data[13]
D1[4]	=	data[12]
D1[3]	=	data[11]
D1[2]	=	data[10]
D1[1]	=	data[9]
D1[0]	=	data[8]
D0[7]	=	data[7]
D0[6]	=	data[6]
D0[5]	=	data[5]
D0[4]	=	data[4]
D0[3]	=	data[3]
D0[2]	=	data[2]
D0[1]	=	data[1]
D0[0]	=	data[0]

GDDR4はデータ並びとして2種類の設定が用意されています。
以下は"By Lane"設定の場合の入出力データの並び方です。



Rx

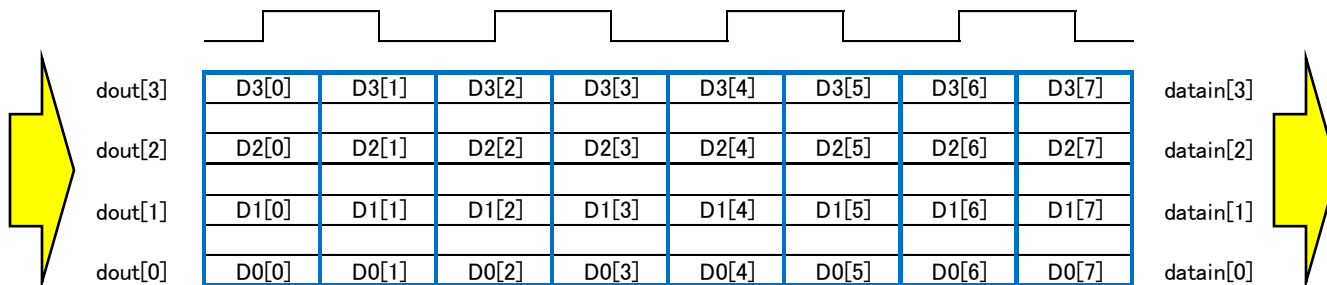
q[31]	=	D3[7]
q[30]	=	D3[6]
q[29]	=	D3[5]
q[28]	=	D3[4]
q[27]	=	D3[3]
q[26]	=	D3[2]
q[25]	=	D3[1]
q[24]	=	D3[0]
q[23]	=	D2[7]
q[22]	=	D2[6]
q[21]	=	D2[5]
q[20]	=	D2[4]
q[19]	=	D2[3]
q[18]	=	D2[2]
q[17]	=	D2[1]
q[16]	=	D2[0]
q[15]	=	D1[7]
q[14]	=	D1[6]
q[13]	=	D1[5]
q[12]	=	D1[4]
q[11]	=	D1[3]
q[10]	=	D1[2]
q[9]	=	D1[1]
q[8]	=	D1[0]
q[7]	=	D0[7]
q[6]	=	D0[6]
q[5]	=	D0[5]
q[4]	=	D0[4]
q[3]	=	D0[3]
q[2]	=	D0[2]
q[1]	=	D0[1]
q[0]	=	D0[0]

【GDDR4入出力データ並び (By Time)】

Tx

D3[7] = data[31]
D2[7] = data[30]
D1[7] = data[29]
D0[7] = data[28]
D3[6] = data[27]
D2[6] = data[26]
D1[6] = data[25]
D0[6] = data[24]
D3[5] = data[23]
D2[5] = data[22]
D1[5] = data[21]
D0[5] = data[20]
D3[4] = data[19]
D2[4] = data[18]
D1[4] = data[17]
D0[4] = data[16]
D3[3] = data[15]
D2[3] = data[14]
D1[3] = data[13]
D0[3] = data[12]
D3[2] = data[11]
D2[2] = data[10]
D1[2] = data[9]
D0[2] = data[8]
D3[1] = data[7]
D2[1] = data[6]
D1[1] = data[5]
D0[1] = data[4]
D3[0] = data[3]
D2[0] = data[2]
D1[0] = data[1]
D0[0] = data[0]

以下は“By Time”設定の場合の入出力データの並び方です。



Rx

q[31] = D3[7]
q[30] = D2[7]
q[29] = D1[7]
q[28] = D0[7]
q[27] = D3[6]
q[26] = D2[6]
q[25] = D1[6]
q[24] = D0[6]
q[23] = D3[5]
q[22] = D2[5]
q[21] = D1[5]
q[20] = D0[5]
q[19] = D3[4]
q[18] = D2[4]
q[17] = D1[4]
q[16] = D0[4]
q[15] = D3[3]
q[14] = D2[3]
q[13] = D1[3]
q[12] = D0[3]
q[11] = D3[2]
q[10] = D2[2]
q[9] = D1[2]
q[8] = D0[2]
q[7] = D3[1]
q[6] = D2[1]
q[5] = D1[1]
q[4] = D0[1]
q[3] = D3[0]
q[2] = D2[0]
q[1] = D1[0]
q[0] = D0[0]

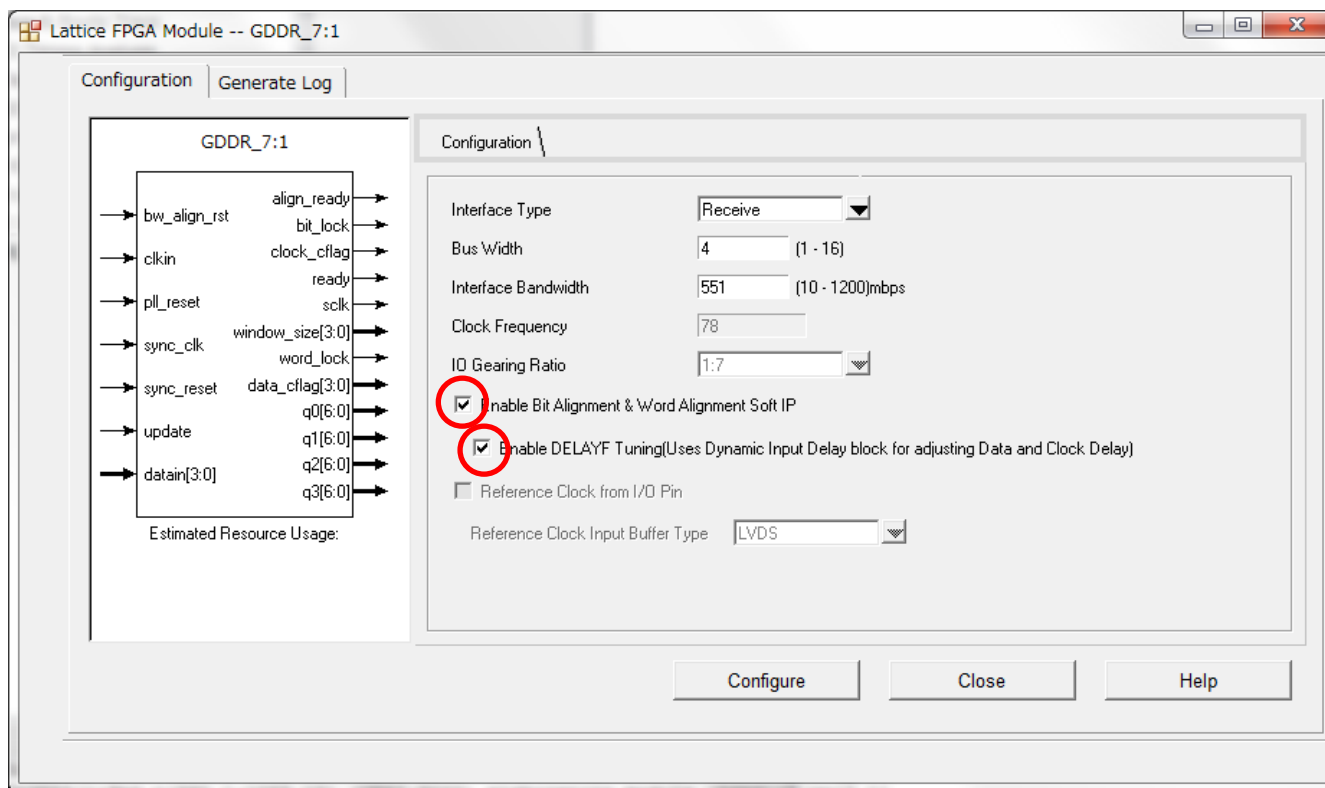
3-4 GDDRサンプル(GDDR71 ECLK)

GDDR71 ECLK (Rx 1:7変換、Tx 7:1変換、クロックは4:3 Dutyで"1100011"の区間にピクセルデータとHsync, Vsync, DEが埋め込まれるインターフェース) の機能説明を以下に記載します。

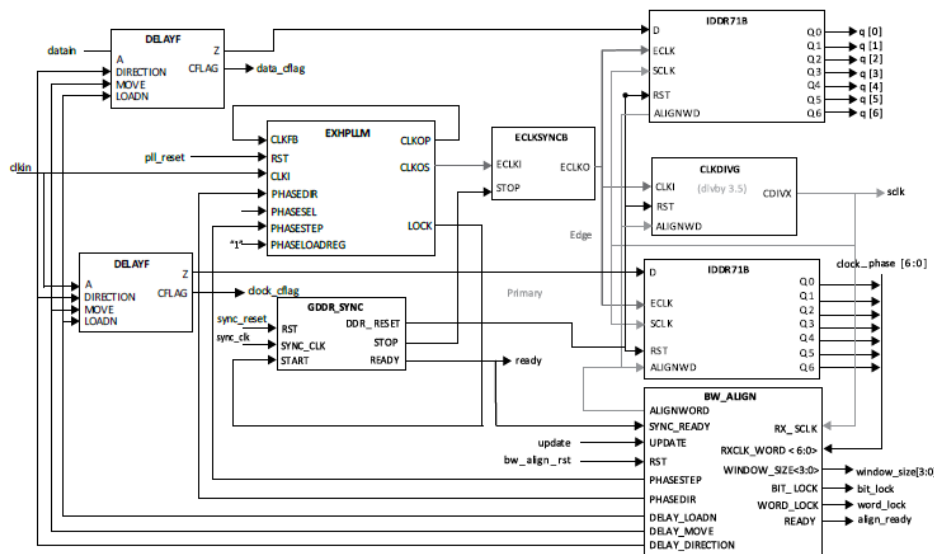
【ポート仕様】

ポート名	I/O方向	説明	コメント
GDDR71_RX.ECLK			
clkin	Input	GDDR71 Rxソースシンクロナスクロック入力。	
datain	Input	GDDR71 Rxシリアルデータ入力。	
pll_reset	Input	PLLリセット入力 (Active High) 。	
bw_align_rst	Input	データアライメントロジックリセット入力 (Active High) 。	
sync_reset	Input	GDDRインターフェースイニシャライズロジックリセット入力 (Active High) 。	
sync_clk	Input	GDDRインターフェースイニシャライズ回路用クロック。ECLKの1/2以下の周波数にする必要があります。	クロックソースはECLKと非同期でも問題ありません。sclkはsync_clkによって動作するイニシャライズシーケンス完了後に出力されるので使用できません。
update	Input	Bit/Wordアライメント開始及びリスタートトリガ信号 (Active High) 。	
ready	Output	GDDR71回路内クロック同期完了通知信号。	
data_cflag	Output	ビットアライメントのためのデータ遅延設定の全設定チェック完了通知信号。	
clock_cflag	Output	ビットアライメントのためのクロックデータ遅延設定の全設定チェック完了通知信号。	
bit_lock	Output	ビットアライメント完了通知信号。	内部通倍クロックがデータをラッチする位置 (位相) を決定したことを示します。
word_lock	Output	ワードアライメント完了通知信号。	"1100011"のクロックタイミングでデータを出し始めたことを示します。
align_ready	Output	アライメントシーケンス完了通知信号。	GDDR71 Rxの正常動作準備完了通知です。
window_size	Output	Valid Window sizeの通知信号。	
sclk	Output	GDDR71 Rxパラレルデータ出力に同期したシステムクロック。	
q	Output	GDDR71 Rxパラレルデータ出力。	
GDDR71_TX.ECLK			
refclk	Input	GDDR71 TxモジュールECLK入力。	ピクセルクロック周波数のx3.5通倍クロックを入力。
data	Input	GDDR71 Tx パラレルデータ入力。	
sync_reset	Input	データアライメントロジックリセット入力 (Active High) 。	
sync_clk	Input	GDDRインターフェースイニシャライズロジック用クロック。ECLKの1/2以下の周波数にする必要があります。	クロックソースはECLKと非同期でも問題ありません。sclkはsync_clkによって動作するイニシャライズシーケンス完了後に出力されるので使用できません。
start	Input	GDDRインターフェースイニシャライズロジックのトリガ信号 (Active High) 。	イニシャライズシーケンスを開始する際にLow -> Highとトグルして下さい。
dout	Output	GDDR71 Txシリアルデータ出力。	
clkout	Output	GDDR71 Txソースシンクロナスクロック出力。	
sclk	Output	GDDR71 Txパラレルデータ入力に同期したシステムクロック。	
ready	Output	GDDR71 Txイニシャライズ完了通知信号。	

GDDR71 Rxデザインはビットアライメント、ワードアライメントがオプションで付属します。
Clarity Designerでモジュールを生成する際に、“Enable Bit Alignment & Word Alignment Soft IP”と
“Enable DELAYF Turning (Uses Dynamic Input Delay block for adjusting Data and Clock Delay)”に
チェックを入れることでアライメント回路が組み込まれます。
(前ページのRx側ポートリスト、及び次ページのブロック図はアライメント回路が組み込まれた状態のものです。)

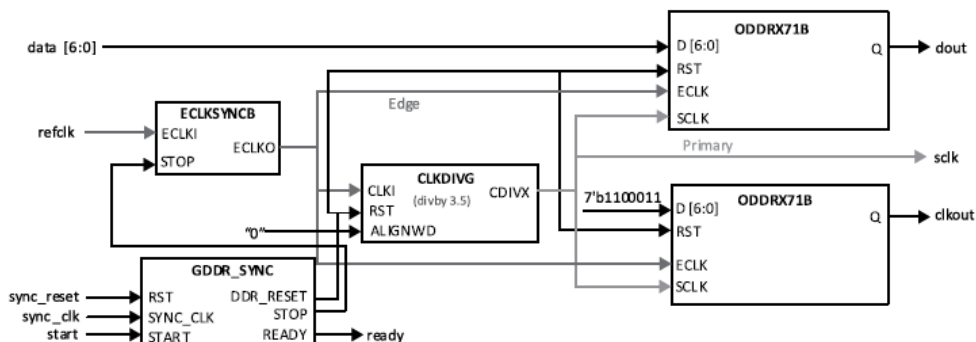


【モジュール構成 : GDDR71_RX.ECLK】



※ GDDR71_RXモジュールからの出力データはsclkに同期して出力されます。そのため、後段回路では必ずGDDR71_RXモジュールから出力されるsclkでデータをラッチして下さい。

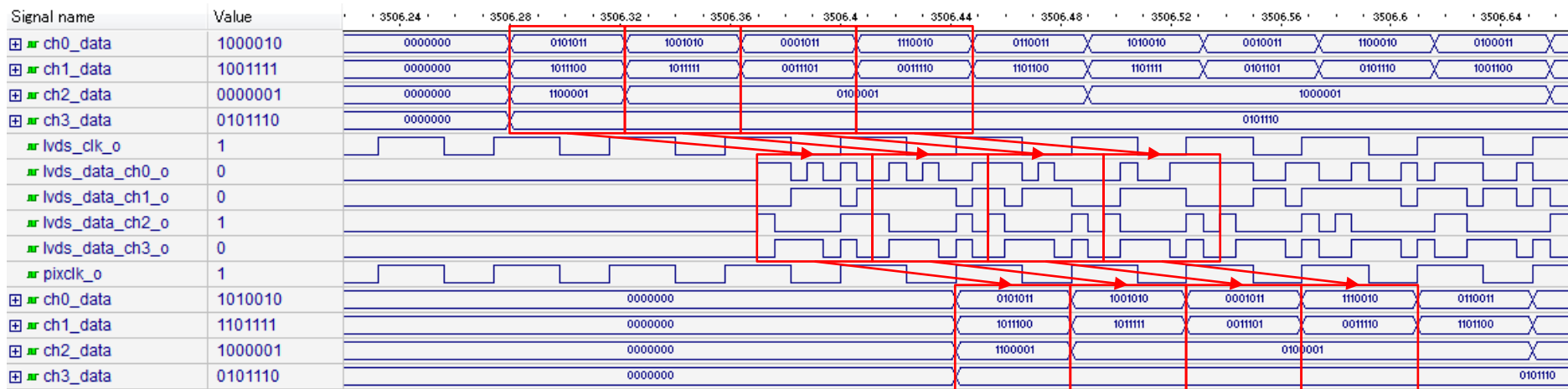
【モジュール構成 : GDDR71_TX.ECLK】



※ GDDR71_TXモジュール内では入力データはsclkでラッチされます。そのため、前段回路では必ずGDDR71_TXモジュールから出力されるsclkに同期させるように設計して下さい。

【GDDR71入出力データ並び】

GDDR71は以下のように“By Lane”の並びでデータ出力されます。“By Lane”についてはP22を参照して下さい。



また、GDDR71の入出力I/FはRGB888フォーマット等になっておらず、28bitパラレルI/Fとなっています。FPD Link等の規格に沿ったデータマッピングにユーザー側で変換する必要があります。

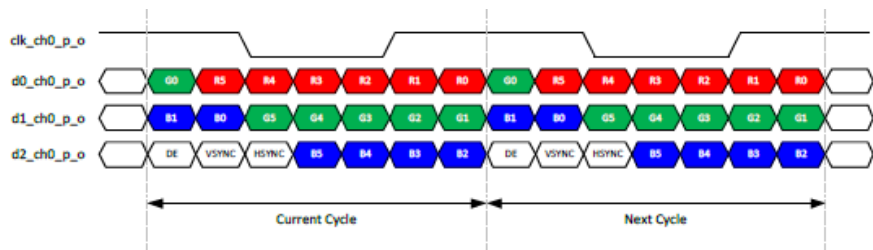


Figure 2.6. FPD-Link Transmit Interface Timing Diagram (RGB666)

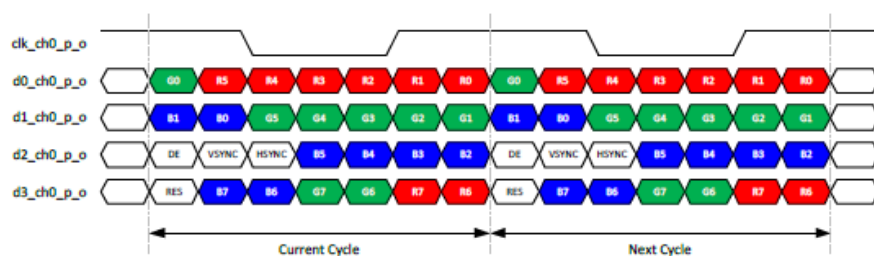


Figure 2.7. FPD-Link Transmit Interface Timing Diagram (RGB888)

CrossLinkはパッケージによってはD-PHYハードマクロを2ch搭載しているものがあります。各chにMIPIDPHY0、 MIPIDPHY1という名前がついており、どちらにアサインするかはDiamondで使用する制約ファイル（.lpfファイル）に以下のようにLOCATE COMP記述でD-PHYハードマクロのインスタンス名をパス記述して指定して下さい。

```
LOCATE COMP " csi2csi_inst/cmos2dphy_inst/dci_wrapper_inst/MIPIDPHYA_inst" SITE "MIPIDPHY0" ;
```

また、アサイン結果はDiamondのMap Reportで確認できます。

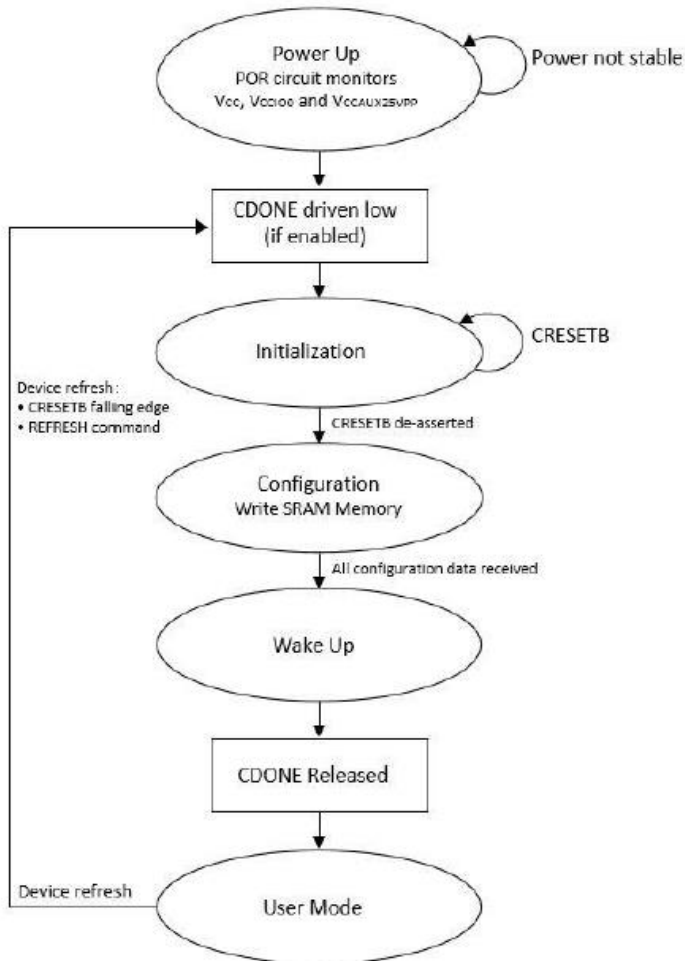
コンフィギュレーション関連

ピン名	使用モード	ピン方向	専用/兼用	説明
CRESET_B	All	入力	専用	Active Lowコンフィグレーションリセット信号。 Low -> Highトグルで再コンフィグレーションを実施。 CRESET_BのLow期間は110ns以上確保する必要があります。
PMU_WKUPN/CDONE	All	双方向	兼用	CDONEはコンフィグ完了を示すOpenDrain出力ピン。 コンフィグ完了でHi-z出力。Power Management Unit (PMU) をスリープモードから復帰させるPMU_WKUPNと兼用。
MISO	MSPI, SSPI	双方向	兼用	MSPIモード時SPIデータ入力。SSPIモード時SPIデータ出力。
MOSI	MSPI, SSPI	双方向	兼用	MSPIモード時SPIデータ出力。SSPIモード時SPIデータ入力。
SPI_SS/CSN/SCL	MSPI, SSPI, I2C	双方向	兼用	MSPIモード時Chip Select出力。SSPIモード時Chip Select 入力。I2C Configurationモード時SCL入力。
SPI_SCK/MCK/SDA	MSPI, SSPI, I2C	双方向	兼用	MSPIモード時SPIクロック出力。SSPIモード時SPIクロック 入力。I2C Configurationモード時SDA入力。

※ MSPI : Master SPI SSPI : Slave SPI

※専用/兼用はユーザーI/Oと兼用かどうかを示しています。兼用ピンであっても可能な限りユーザーI/Oとして使用しないことを推奨しています。

※CrossLinkは出荷状態でMSPI、SSPIの両方がEnableになっています。この状態はSPI FlashからのコンフィグレーションとPCからDiamond Programmerを用いたコンフィグレーション、及びCPU等からのSSPIコンフィグレーションが可能な状態です。Diamondの設定ではMSPIとSSPIを同時にEnableにすることはできず、上記状態は出荷時状態の設定でのみ維持されます。CrossLinkの内蔵NVCMはワンタイムプログラムのため、一度設定を変更したデータを書き込むと二度と設定を変更したり、初期状態に戻したりすることができません。設定を変更したデータを書き込む際には注意して下さい。



1. 電源立ち上がり後、Power On Resetがかかり、CDONEがLowにドライブされます。
2. その後イニシャライゼーションが実行され、CRESET_BのHighを検出するとコンフィグレーションに移ります。CRESET_BをLowに保持することでコンフィグレーションの開始を遅らせることができます。
3. 全てのコンフィグレーションデータがSRAM領域に展開された後、CDONEがLow -> Hi-zになり、ロジック動作を開始します。CDONEを外部プルアップしておくことでコンフィグレーション完了をモニターできます。
4. デバイス動作中にCRESET_Bの立下りエッジ、またはSSPIやI2CによるRefresh commandを検出すると、CDONEがLowに落ち、デバイスがイニシャライゼーションから再起動します。

Master SPIコンフィグレーションで外部SPI Flashに格納するコンフィグデータ容量、及びSlave SPI コンフィグレーション時にHostのROM領域に格納するコンフィグデータ容量は以下の通りです。

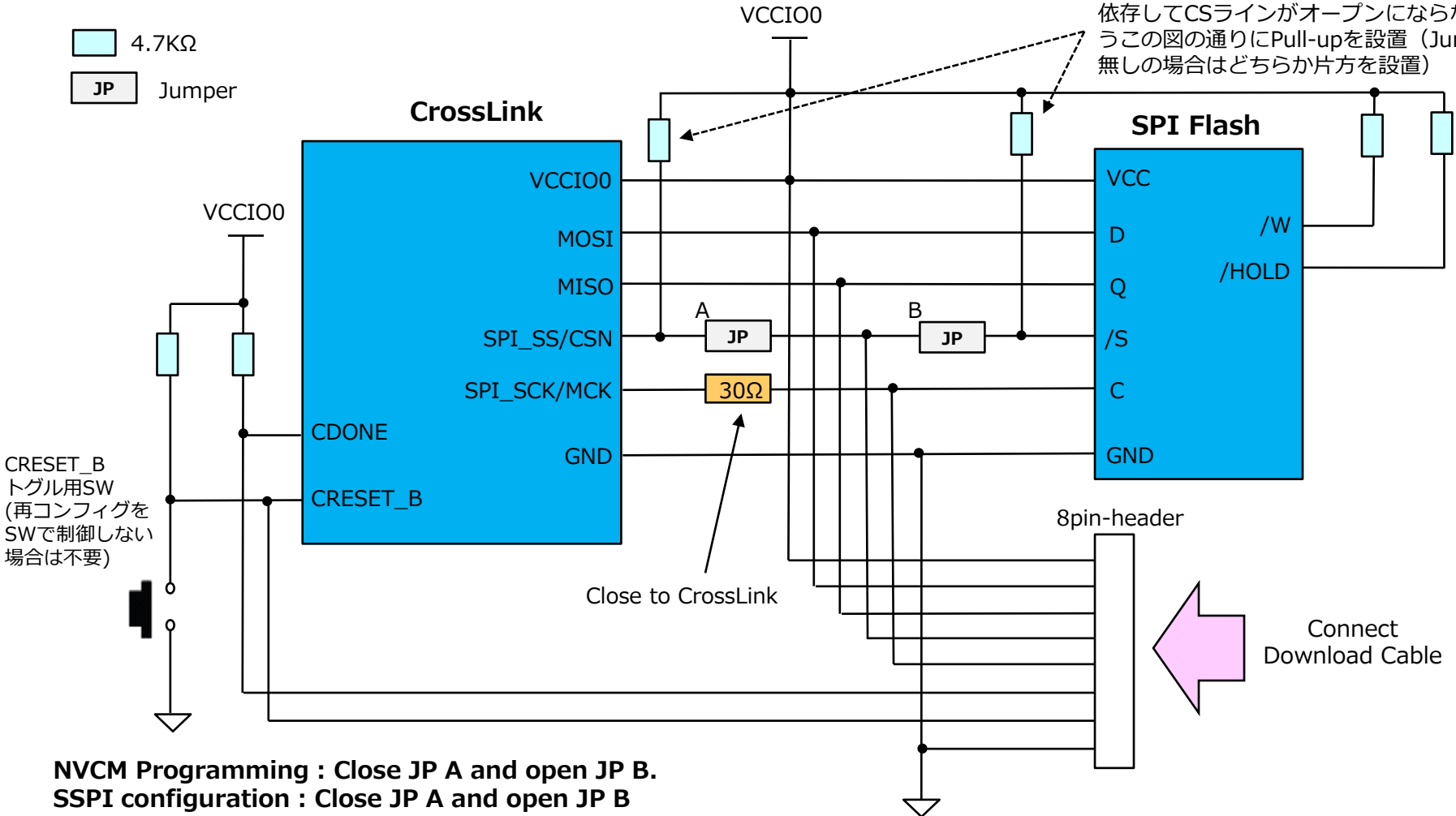
Device	Bitstream Size Without Pre-Initialized EBR	Bitstream Size With Maximum Number of Pre-Initialized EBR	Units
LIF-MD6000	1.24	1.59	Mb

4-4 コンフィグレーション回路

CSラインのPull-up抵抗。
A, B Jumperありの場合はJumperの設定に
依存してCSラインがオープンにならないよ
うこの図の通りにPull-upを設置 (Jumper
無しの場合はどちらか片方を設置)

4.7KΩ

JP Jumper



CRESET_B
トグル用SW
(再コンフィグを
SWで制御しない
場合は不要)

Close to CrossLink

8pin-header

Connect
Download Cable

- NVCM Programming : Close JP A and open JP B.**
- SSPI configuration : Close JP A and open JP B**
- SPI Flash Programming : Close JP B and open JP A.**
- MSPI Configuration : Close JP A and B.**

デバイスとDownload Cableの接続は以下の通りです。

[NVCM programming]

CrossLink		Download Cable
MOSI	→	TDI
MISO	→	TDO
SPI_SS/CSN	→	ISPEN
SPI_SCK/MCK	→	TCK
CDONE	→	NC
CRESET_B	→	TRST
Vccio0	→	Cable Vcc
GND	→	Cable GND

[SPI Flash Programming]

SPI Flash		Download Cable
D	→	TDI (SPI Flash SI)
Q	→	TDO (SPI Flash SO)
/S	→	ISPEN
C	→	TCK
CDONE	→	NC
CRESET_B	→	NC
Vcc	→	Cable Vcc
GND	→	Cable GND

[SSPI Configuration]

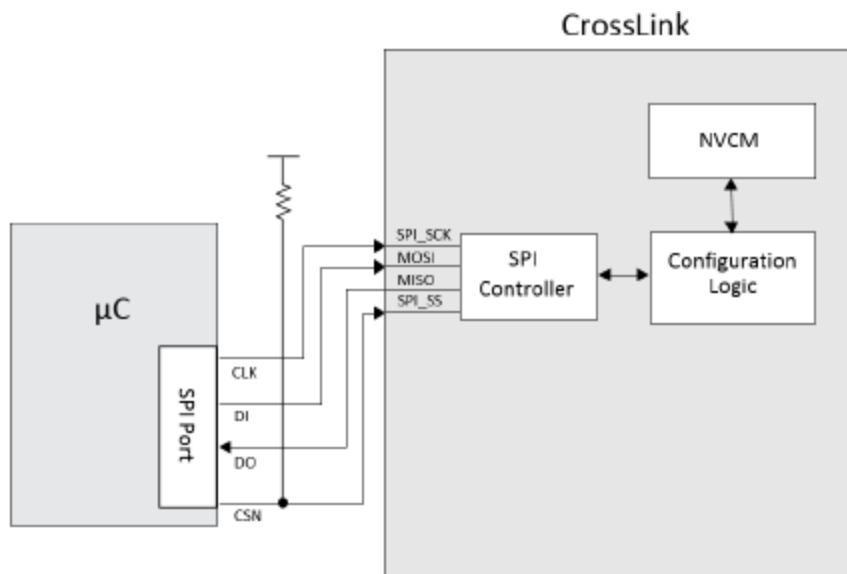
CrossLink		Download Cable
MOSI	→	TDI
MISO	→	TDO
SPI_SS/CSN	→	ISPEN
SPI_SCK/MCK	→	TCK
CDONE	→	NC
CRESET_B	→	TRST
Vccio0	→	Cable Vcc
GND	→	Cable GND

'NC' : Not connected

CrossLinkのコンフィグレーションモードとして、MCU等のホストからSPIやI2Cインターフェースでコンフィグレーションを行うモードが用意されています。ここでは例としてSPIインターフェースでコンフィグレーションを行うモードを説明します。

SPIでコンフィグレーションを行う際には、CrossLinkのSlave SPIモードを設計ツールDiamondでEnableに設定するか、または出荷状態からコンフィグレーションモードを変更していないことが条件になります。MCU等のホストのROMにコンフィグレーションデータを持たせ、所定のフローに従ってCrossLinkにアクセスし、コンフィグレーションデータを送ります。

次ページ以降にSlave SPIコンフィグレーションのサンプルフローを示します。



4-5 Slave SPI Configuration

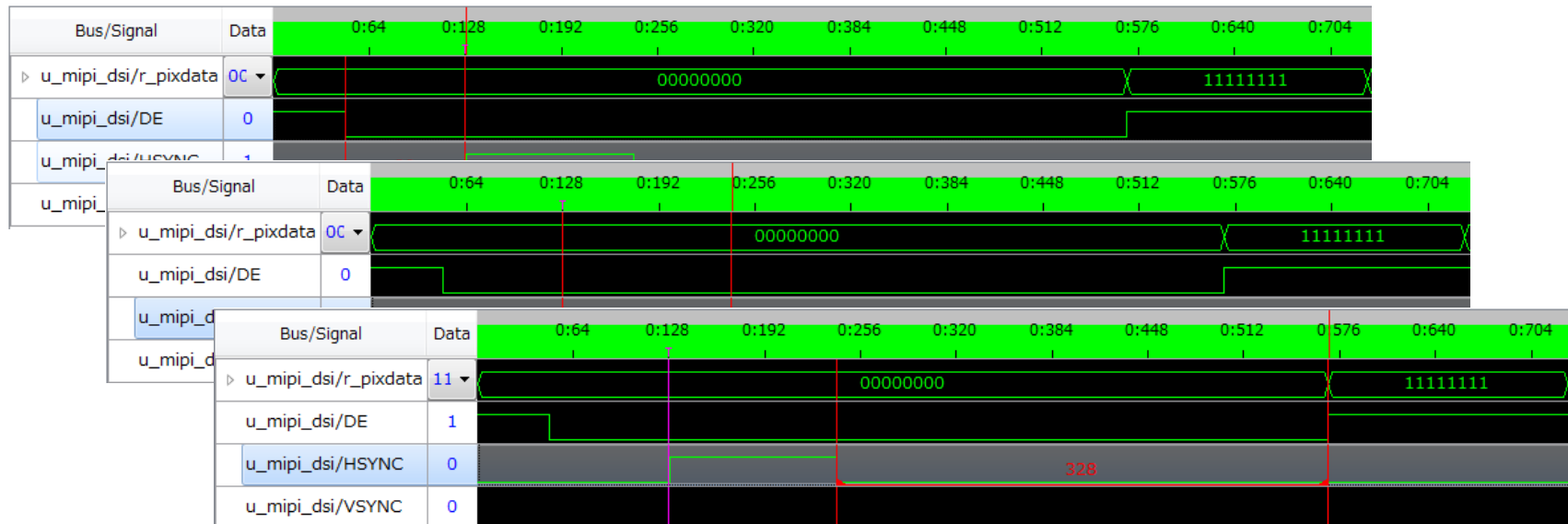
Crosslink Slave-SPI SRAM Configuration Flow

Step#	Configuration Step	Shift In			Shift Out		Description	
		Signal	Clocks	Value	Signal	Value		
1	Initialize	CRESETB		LOW				
		Wait 1000 ms						
		SN		LOW				
		SI	40	0xFF A4 C6 F4 8A			Shift in Key activation	
		SN		HIGH				
		CRESETB		HIGH				
2	Delay	Wait 10 ms						
3	Check the IDCODE (Optional)	SN		LOW				
		SI	32	0xE0 00 00 00			Shift in IDCODE_PUB(0xE0) opcode	
		SI	32	0x00 00 00 00	SO	Device_ID	Shift out Device ID and compare to expected value	
		SN		HIGH				
4	Enable Programming Mode	SN		LOW				
		SI	32	0xC6 00 00 00			Shift in ISC ENABLE(0xC6) instruction	
		SN		HIGH				
5	Delay	Wait 1 ms						
6	Erase the device	SN		LOW				
		SI	32	0x0E 00 00 00			Shift in ISC ERASE(0x0E) instruction	
		SN		HIGH				
7	Delay	Wait 200 ms						

4-5 Slave SPI Configuration

8	Program Fuse Map	SN		LOW			
		SI	32	0x46 00 00 00			Shift in LSC_INIT_ADDRESS(0x46) instruction
		SN		HIGH			
		SN		LOW			
		SI	32	0x7A 00 00 00			Shift in LSC_BITSTREAM_BURST(0x7A) instruction
		SI	xxx	Full Bitstream			Shift in bitstream (.bit) generated by Diamond.
		SN		HIGH			
9	Delay	Wait 10 ms					
10	Read the status bit	SN		LOW			
		SI	32	0x3C 00 00 00			Shift in LSC_READ_STATUS(0x3C) instruction
		SI	32	0x00 00 00 00	S0	Status Register Value	Expect value from S0: 0x00000100 with the Mask: 0x00003100 [Bit-8 DONE = 1, Bit-12 Busy = 0, Bit-13 Fail = 0]
		SN		HIGH			
11	Exit Programming Mode	SN		LOW			
		SI	32	0x26 00 00 00			Shift in ISC DISABLE(0x26) instruction
		SN		HIGH			
		Wait 200 ms					
		SN		LOW			
		SI	32	0xFF FF FF FF			Shift in NO-OP(0xFF) instruction
SN		HIGH					

内部波形観測ツール“Reveal”使用時の注意



CrossLinkとPCをDownload Cableで接続し、動作中に内部波形の観測を行うことができます。内部波形観測にはRevealというJTAG I/Fのツールを使用しますが、CrossLinkはJTAG I/Fを持っていないため、JTAG回路をエミュレートした回路を埋め込み、汎用I/OをJTAGピンに見立ててRevealを動作させます。

JTAG I/FとしてTDO、TDI、TMS、TCKの4本が必要になります。

CrossLinkはMulti voltage Inputに対応していないため、これら4本の信号をアサインするI/OのBank Vccioは同じにするよう回路図設計時から注意して下さい。

また、TCKはクロック入力専用ピン（PCLKTx_x）にアサインする必要がありますのでご注意ください。（クロック入力専用ピン以外にアサインするとDiamondのPlace&Routeでエラーになります。）

Revealの使用方法については別途弊社FAEにお問い合わせ下さい。

Date	Revision	Old-page	New-Page	Change information	Updated by
2017/3/30	1.0			First Revision	W.Nakatsuka
2017/6/20	1.1	P24	P24	GDDR71_TX.ECLKのrefclkの説明を修正	W.Nakatsuka
2017/7/19	1.2	P40	P40	Reveal用JTAG_TCKのアサイン制約を追記	W.Nakatsuka
2020/9/15	1.3	P15, P16	P15, P16	P15 GR_PCLK[Bank]0の説明を修正 P16 差動ペアの注意書きを修正	W.Nakatsuka
2020/10/2	1.4	P34	P34	CSラインのプルアップの記載を追加	W.Nakatsuka