

1 先頭サンプルの時刻(トリガ(t=0)基準)の算出方法

算出方法の確認のために、以下の波形を測定した。

- ・デモジュレータのトリガ信号
- ・デモジュレータのクロック信号
- ・モジュールの Export された ai/StartTrigger 信号

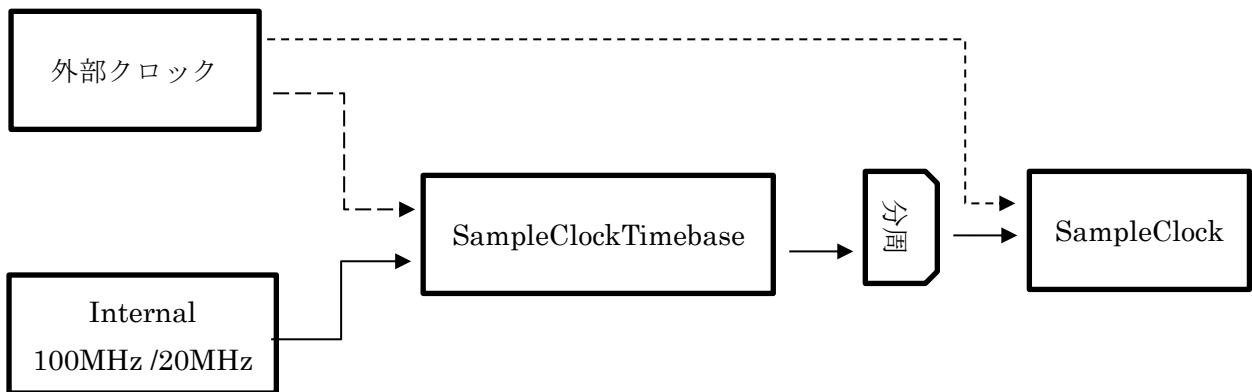
測定地点の波形は、モジュールの動作クロック(100MHz)の1クロック(10ns)遅延されている。

- ・モジュールの Export された ai/SampleClock 信号

測定地点の波形は、モジュールの動作クロック(100MHz)の1クロック(10ns)遅延されている。

測定結果については、第2章以降に記述した。

以下は、タイミングに関与するクロックの関連図である。



用語説明

(1) StartClockDelay

SampleClockTimebase の使用時に、SampleClock の開始をスタートトリガ(ai/StartTrigger)から遅延させるクロック数(tick 数)である。モジュール設定値である。

※ PXIe-6378 モジュール 16CH 16bit MAX3.5MS/s/ch X シリーズ

1. 1 Internal 100MHz/ 20MHz

SampleClockTimebase に InternalClock(100MHz) または、InternalClock(20MHz)を設定した場合である。

サンプリング周期= InternalClock 周期×内部クロック分周数(Div)

(1) Continuous, Event(Finite)モード

先頭サンプルの時刻 = ジッタ + 処理時間(100ns) + SampleClockDelay × InternalClock 周期

- ・トリガ検出時のジッタが最大 InternalClock 周期分発生する。
- ・処理時間はトリガ検出時間等である。
- ※SampleClockEdge 設定は無効と思われる。

(2) Event モード (プリサンプル n>0)

先頭サンプルの時刻 = -n × サンプリング周期 + ジッタ

- ・サンプリングはトリガ前に不定タイミングで開始されているので、最大、サンプル周期のジッタが発生する。

1. 2 外部クロック (ソース: デモジェネレータ)

1. 2. 1 SampleClockTimebase に外部クロックを設定した場合

サンプリング周期 = 外部クロック周期(sec) × クロック分周数(Div)

(1) Continuous, Event(Finite)モード

先頭サンプルの時刻 = StartClockDelay × 外部クロック周期 + 処理時間 70ns

※SampleClockEdge 設定は無効と思われる。

(2) Event モード (プリサンプル n>0)

先頭サンプルの時刻 = -n × サンプリング周期 + ジッタ

サンプリングはトリガ前に不定タイミング(分周のどこか)で開始されているので、最大、サンプル周期分のジッタが発生する。

※SampleClockEdge 設定は無効と思われる。

1. 2. 2 SampleClock に外部クロックを設定した場合

サンプリング周期 = 外部クロック周期

(1) Continuous, Event(Finite)モード

- SampleClockEdge=Rising 時

先頭サンプルの時刻 = 外部クロック周期 + 処理時間 70ns

外部クロック周期分は、トリガ検出が第1クロックの Rising に間に合わないためである。

- SampleClockEdge=Falling 時

先頭サンプルの時刻 = (外部クロック周期 / 2) + 処理時間 70ns

(2) Event モード (プリサンプル n>0)

- SampleClockEdge=Rising 時

先頭サンプルの時刻 = -n × サンプリング周期 + 処理時間 70ns

- SampleClockEdge=Falling 時

先頭サンプルの時刻 = -n × サンプリング周期 - (外部クロック周期 / 2) + 処理時間 70ns

2 測定結果

波形線は、上から順に、以下の通りである。

- ・黄色：外部トリガ(デモジュレータのトリガ)信号
- ・緑色：モジュールの Export された ai/StartTrigger 信号
- ・赤色：モジュールの Export された ai/SampleClock 信号
- ・青色：外部サンプルクロック(デモジュレータのクロック)信号

測定パターンを、以下に示す。

| 番号 | クロックソース | 分周 | モード | Start Clock Delay (ticks) | Sample Clock Edge |
|-----|-----------------|----|--------------|---------------------------|-------------------|
| 1-1 | Internal 100MHz | 40 | Continuous | 2 | (Rising) |
| 1-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 1-3 | ↑ | ↑ | ↑ | 10 | (Rising) |
| 1-4 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 2-1 | Internal 20MHz | 8 | ↑ | 2 | (Rising) |
| 2-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 2-3 | ↑ | ↑ | ↑ | 10 | (Rising) |
| 2-4 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 3-1 | Internal 100MHz | 40 | Finite | 2 | (Rising) |
| 3-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 3-3 | ↑ | ↑ | ↑ | 10 | (Rising) |
| 3-4 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 4-1 | Internal 20MHz | 8 | ↑ | 2 | (Rising) |
| 4-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 4-3 | ↑ | ↑ | ↑ | 5 | (Rising) |
| 4-4 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 5-1 | Internal 100MHz | 40 | PreSamples 5 | (2) | (Rising) |
| 5-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 6-1 | Internal 20MHz | 8 | ↑ | ↑ | (Rising) |
| 6-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 7-1 | External 10MHz | 4 | Continuous | 2 | (Rising) |
| 7-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 7-3 | ↑ | ↑ | ↑ | 5 | (Rising) |
| 7-4 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 8-1 | ↑ | ↑ | Finite | 2 | (Rising) |
| 8-2 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 8-3 | ↑ | ↑ | ↑ | 5 | (Rising) |
| 8-4 | ↑ | ↑ | ↑ | ↑ | (Falling) |
| 9-1 | ↑ | ↑ | PreSamples 5 | (2) | Rising |
| 9-2 | ↑ | ↑ | ↑ | ↑ | Falling |

| 番号 | クロックソース | 分周 | モード | Start Clock Delay (ticks) | Sample Clock Edge |
|------|-----------------|----|-------------|---------------------------|-------------------|
| 10-1 | External 2.5MHz | 1 | Continuous | (2) | Rising |
| 10-2 | ↑ | ↑ | ↑ | ↑ | Falling |
| 11-1 | ↑ | ↑ | Finite | ↑ | Rising |
| 11-2 | ↑ | ↑ | ↑ | ↑ | Falling |
| 12-1 | ↑ | ↑ | PreSamples5 | ↑ | Rising |
| 12-2 | ↑ | ↑ | ↑ | ↑ | Falling |

考察

(1) 分周器を利用する時

- ・ ai/SampleClock 波形のパルス幅は常に約 40ns である。
- ・ SampleClockEdge 設定にかかわらず、ai/SampleClock の Rising がサンプリングタイミングと思われる。

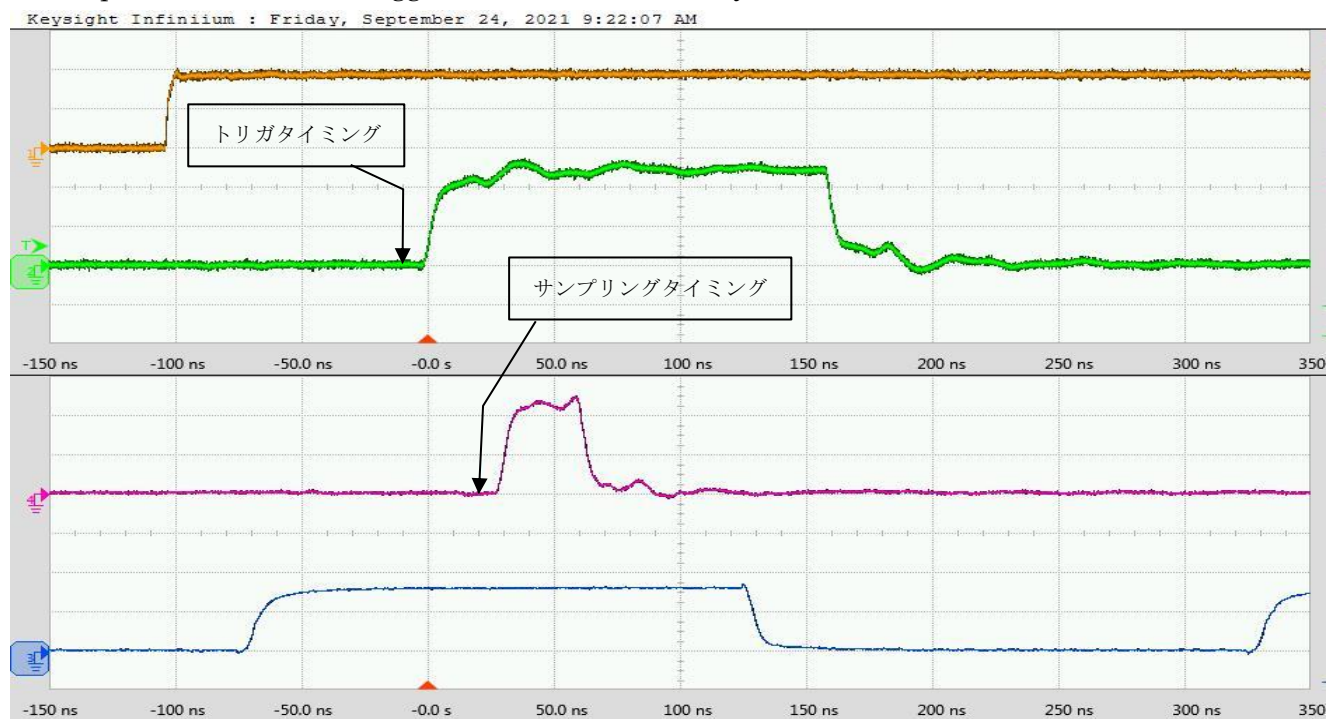
(1) Internal 100MHz, Div200, Continuous
SampleClockTimebaseActiveEdge=Rising である。

(1-1) Internal 100MHz, Div40, Continuous, Start Clock Delay 2ticks, Sample Clock Rising

ai/StartTrigger : SampleClockTimebase の同期ジッタ(最大 10ns)

処理時間 90ns+ 表示上遅延 10ns

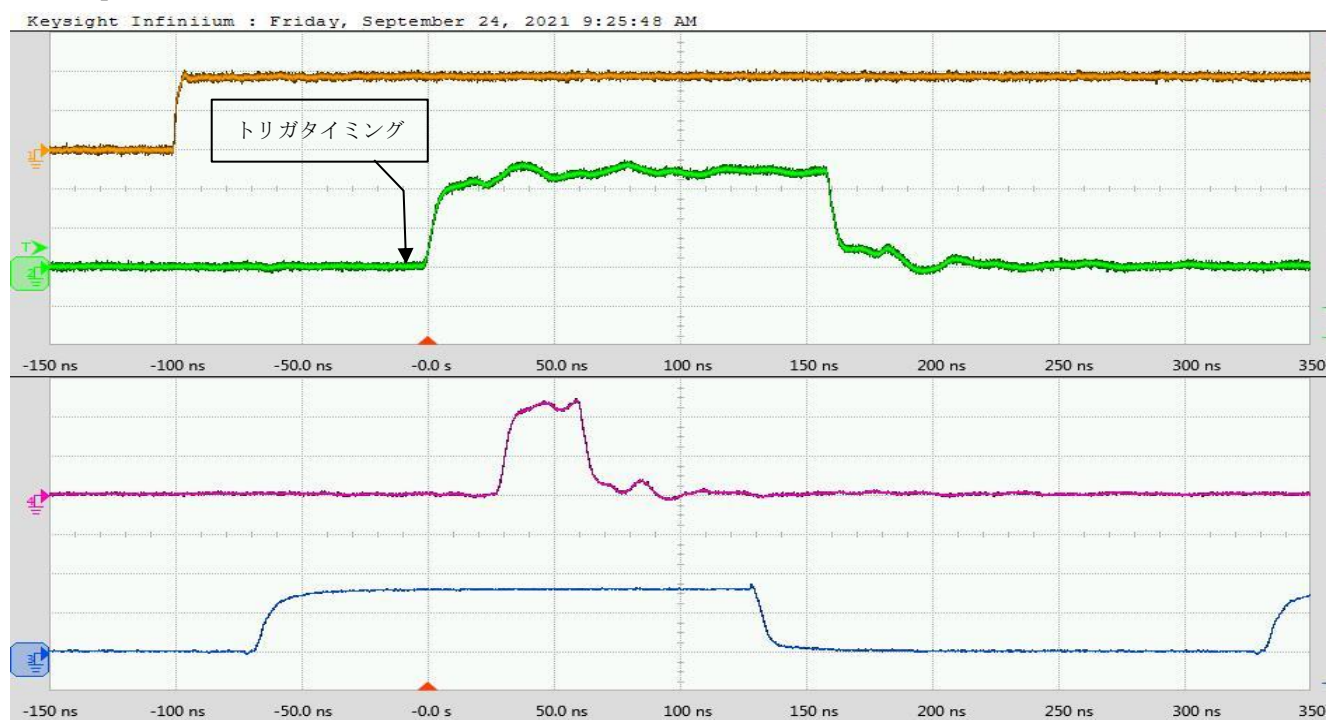
ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。30ns (2ticks+10ns)



(1-2) (1-1) Sample Clock Rising → Falling

ai/StartTrigger : (1-1)と同じである。

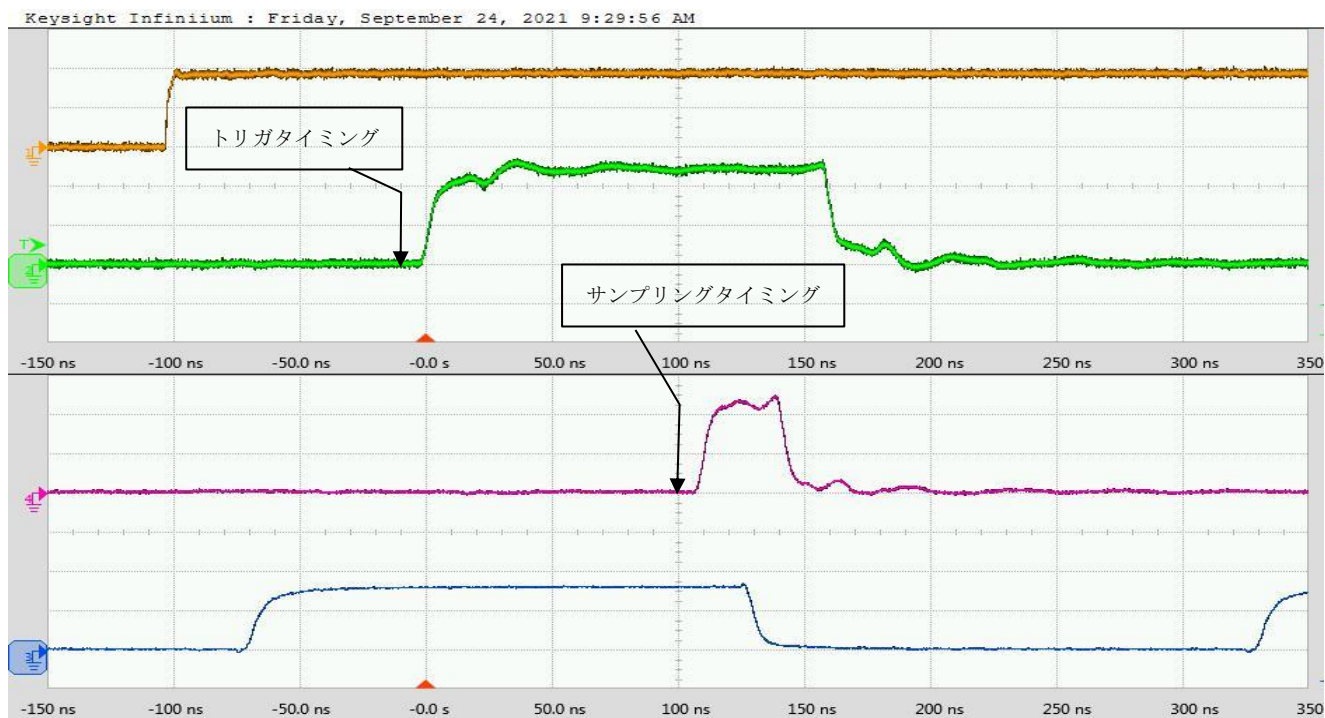
ai/SampleClock : (1-1)と同じである。



(1-3) (1-1) Sample Clock Delay 2ticks → 10ticks

ai/StartTrigger : (1-1)と同じである。

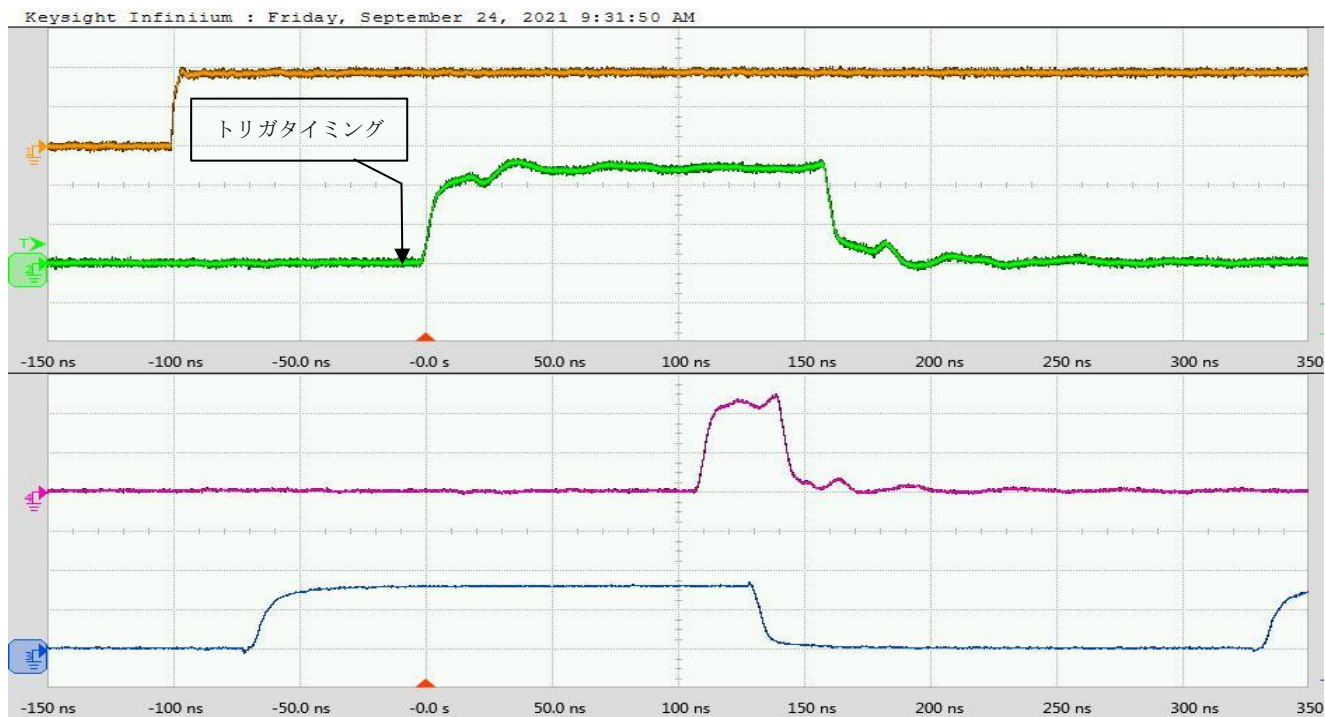
ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。110ns (10ticks+10ns)



(1-4) (1-3) Sample Clock Rising → Falling

ai/StartTrigger : (1-1)と同じである。

ai/SampleClock : (1-3)と同じである。



(2) Internal 20MHz, Div8, Continuous

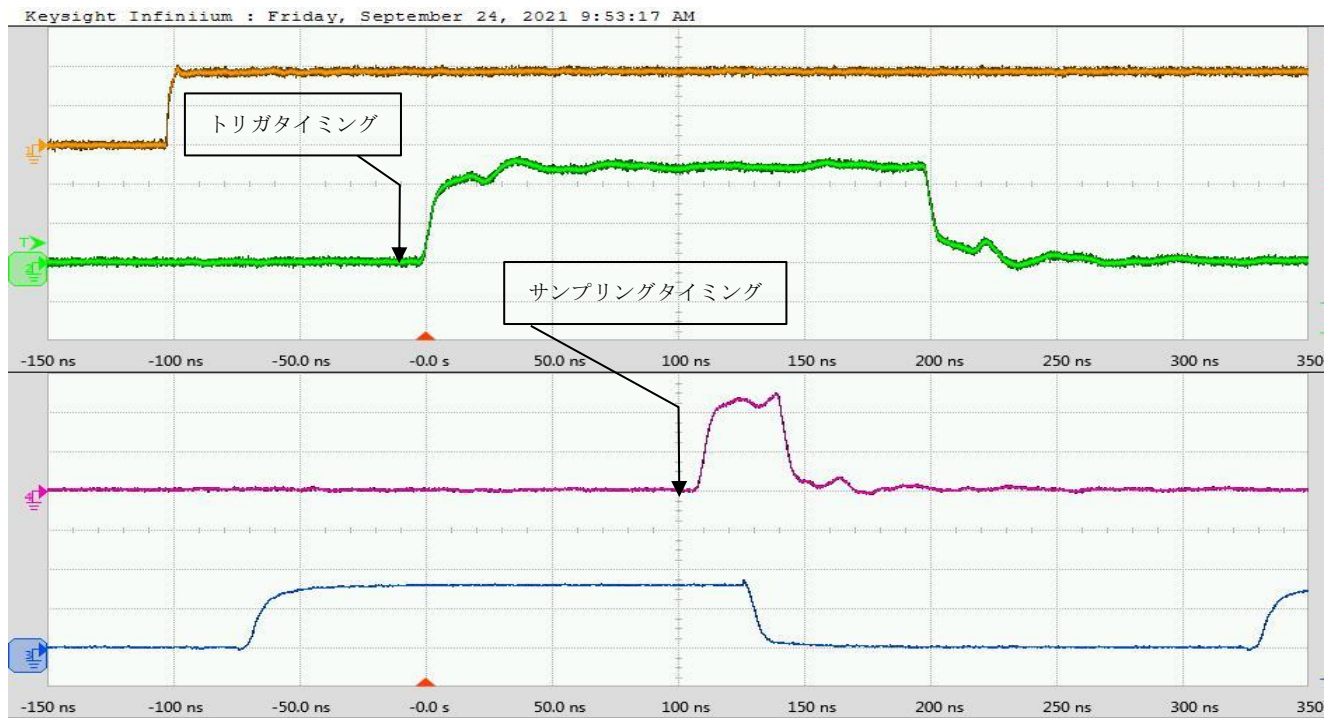
SampleClockTimebaseActiveEdge=Rising である。

(2-1) Internal 20MHz, Div8, Continuous, Start Clock Delay 2ticks, Sample Clock Rising

ai/StartTrigger : SampleClockTimebase の同期ジッタ(最大 50ns)

処理時間 90ns+ 表示上遅延 10ns

ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。110ns (2ticks+10ns)



(2-2) (2-1) Sample Clock Rising → Falling

ai/StartTrigger : (2-1)と同じである。

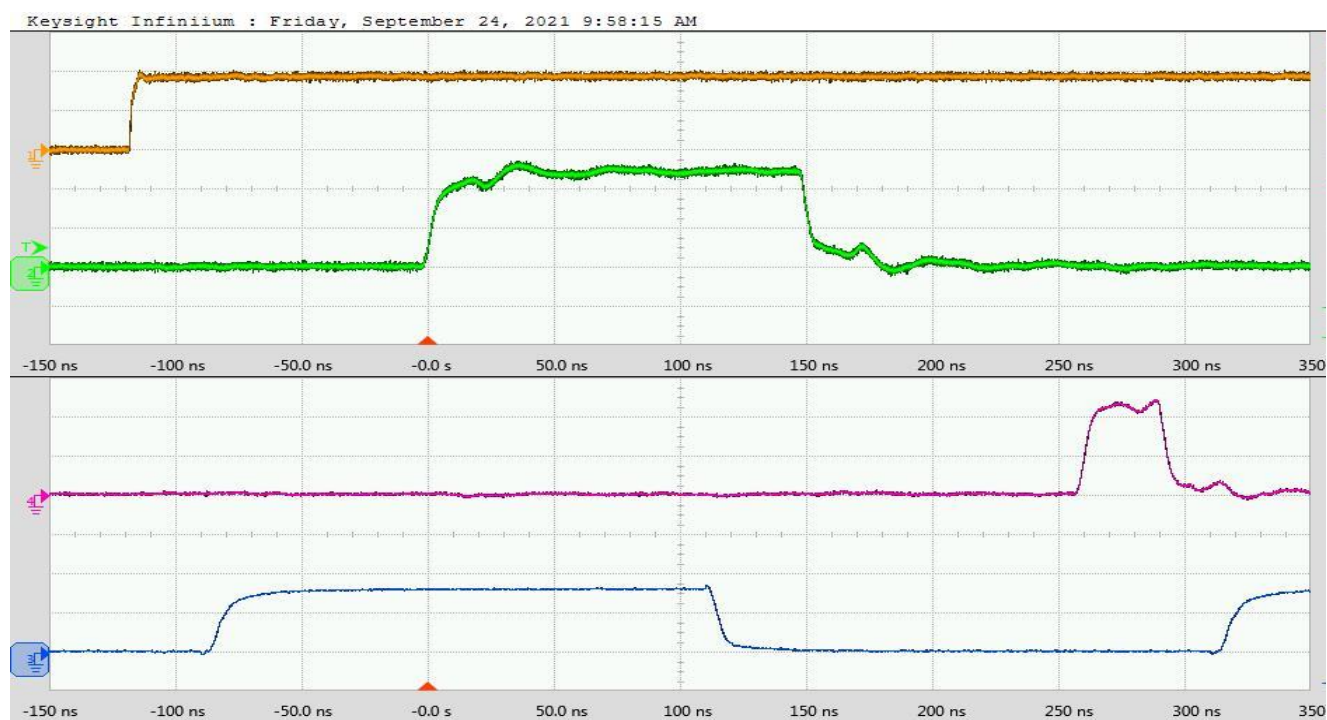
ai/SampleClock : (2-1)と同じである。



(2-3) (2-1) Sample Clock Delay 2ticks → 5ticks

ai/StartTrigger : (2-1)と同じである。

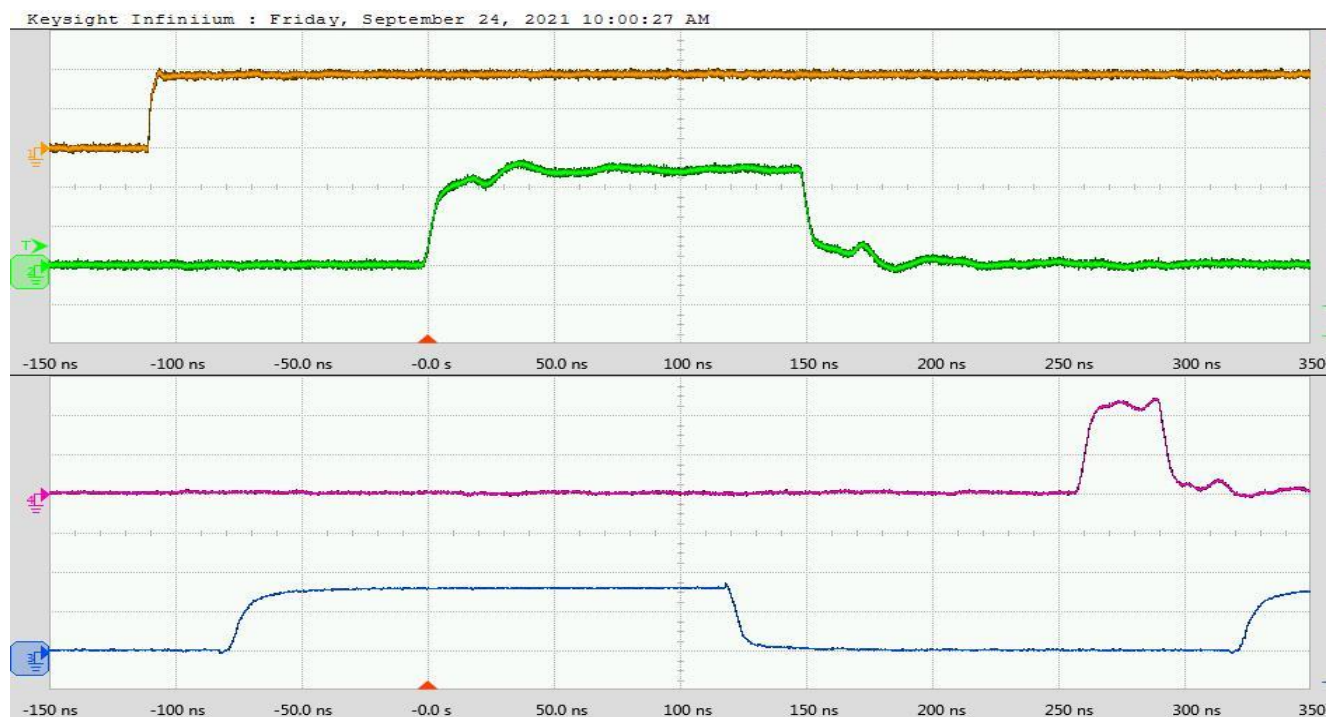
ai/SampleClock : ai/StartTrigger から Start Clock Delay のずれ。 260ns (5ticks+10ns)



(2-4) (2-3) Sample Clock Rising → Falling

ai/StartTrigger : (2-1)と同じである。

ai/SampleClock : (2-3)と同じである。



(3) Internal 100MHz, Div40, Finite

SampleClockTimebaseActiveEdge=Rising である。

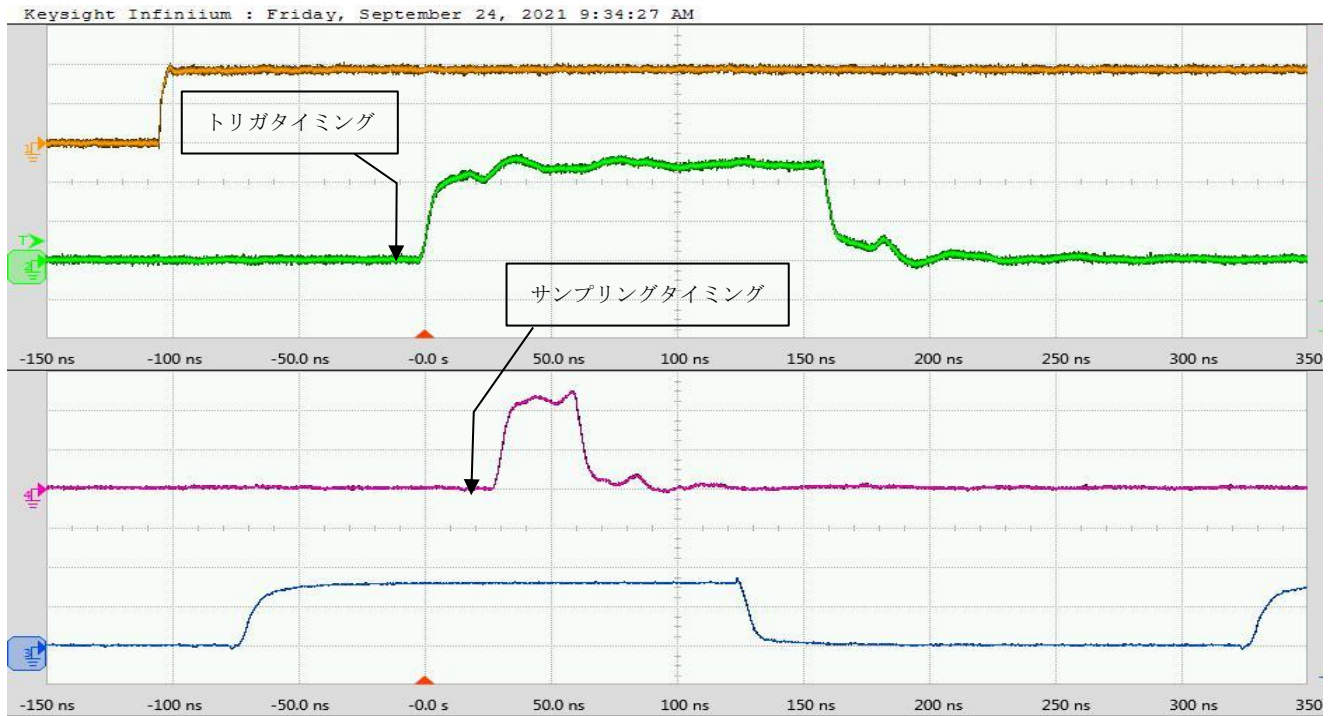
(1) Continuous モードと同じ結果と判断する。

(3-1) Internal 100MHz, Div40, Finite, Start Clock Delay 2ticks, Sample Clock Rising

ai/StartTrigger : SampleClockTimebase の同期ジッタ(最大 10ns)

処理時間 90ns+ 表示上遅延 10ns

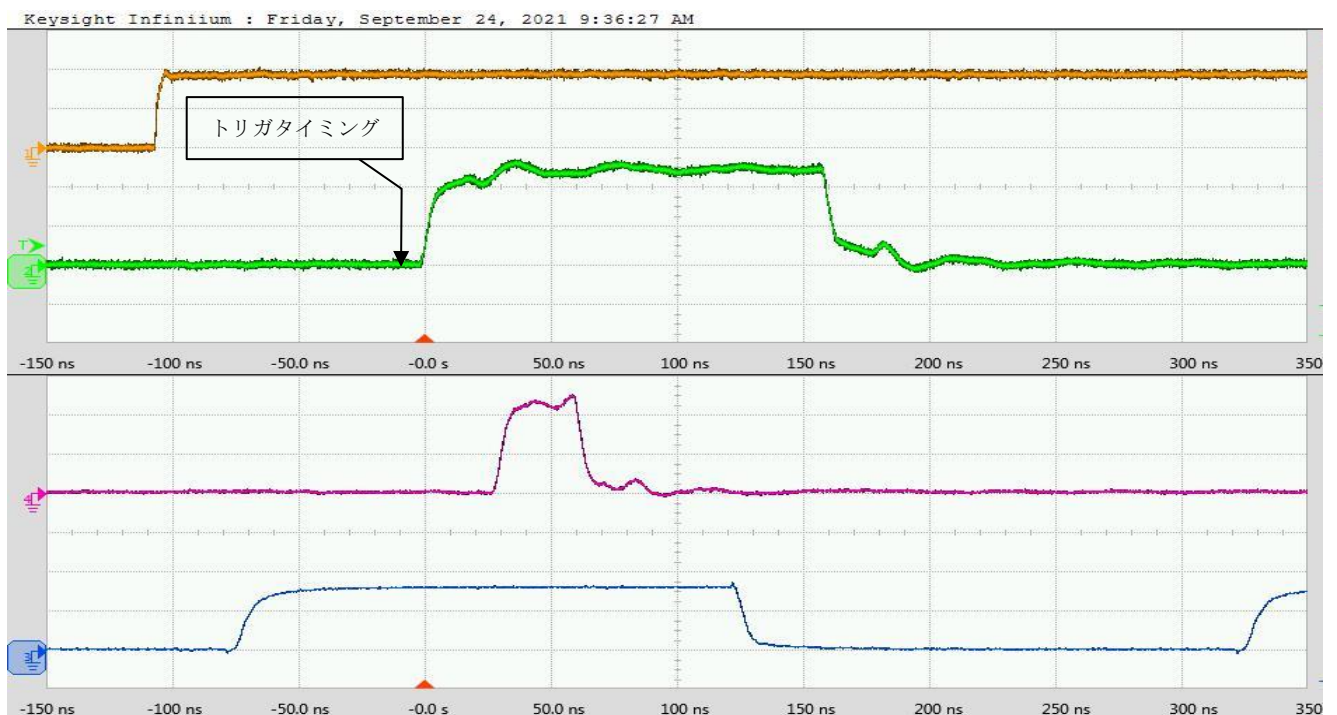
ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。 30ns (2ticks+10ns)



(3-2) (3-1) Sample Clock Rising → Falling

ai/StartTrigger : (3-1)と同じである。

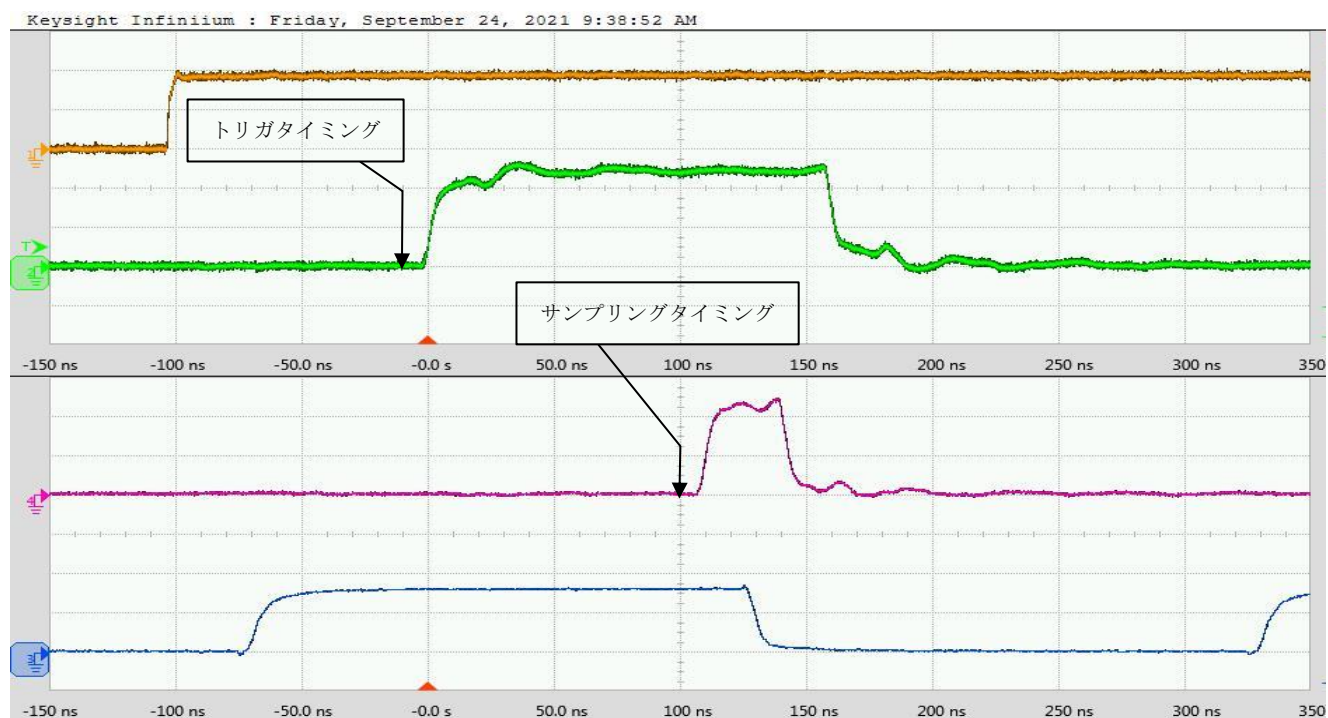
ai/SampleClock : (3-1)と同じである。



(3-3) (3-1) Start Clock Delay 2ticks → 10ticks

ai/StartTrigger : (3-1)と同じである。

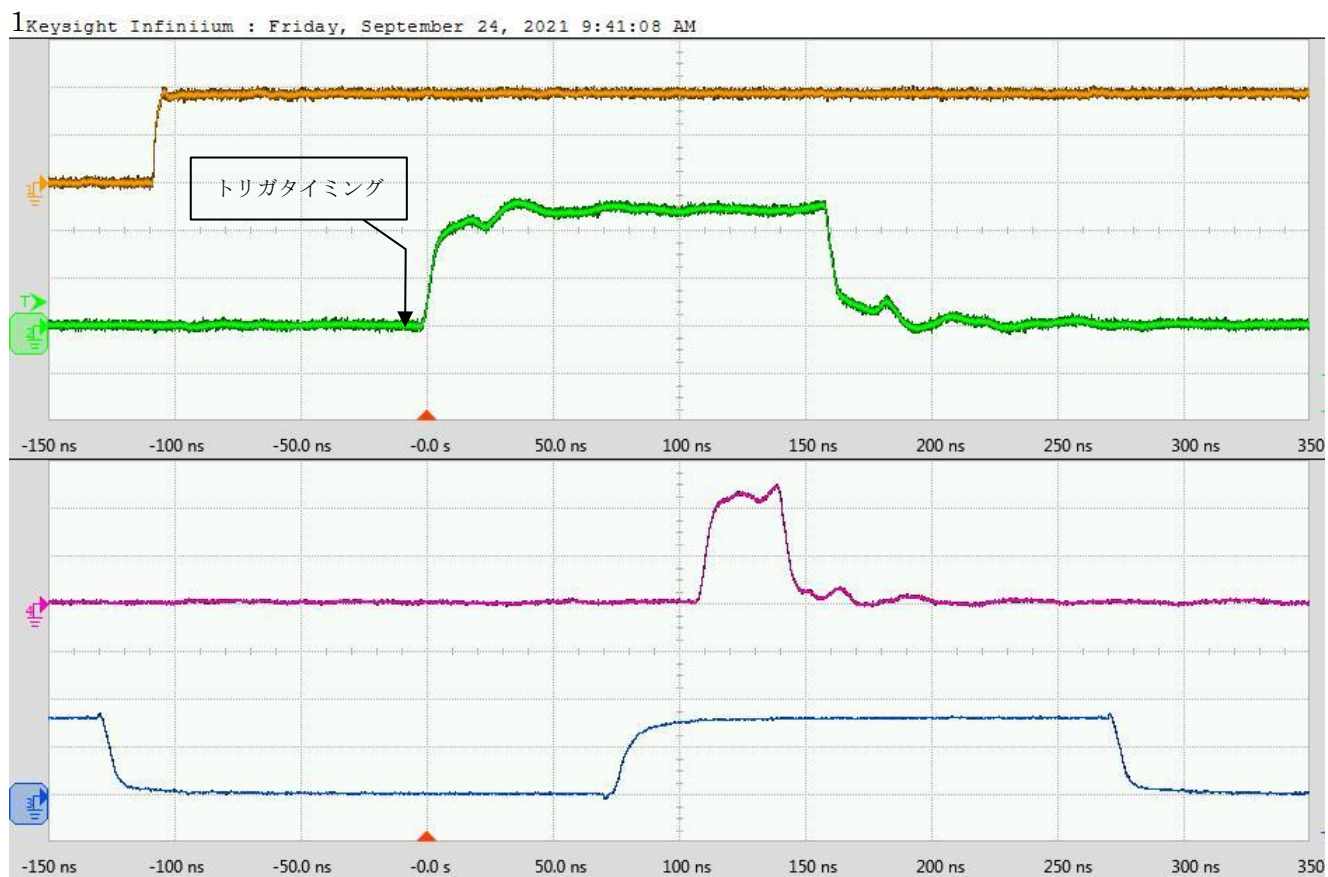
ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。110ns (10ticks+10ns)



(3-4) (3-3) Sample Clock Rising → Falling

ai/StartTrigger : (3-1)と同じである。

ai/SampleClock : (3-3)と同じである。



(4) Internal 20MHz, Div8, Finite

SampleClockTimebaseActiveEdge=Rising である。

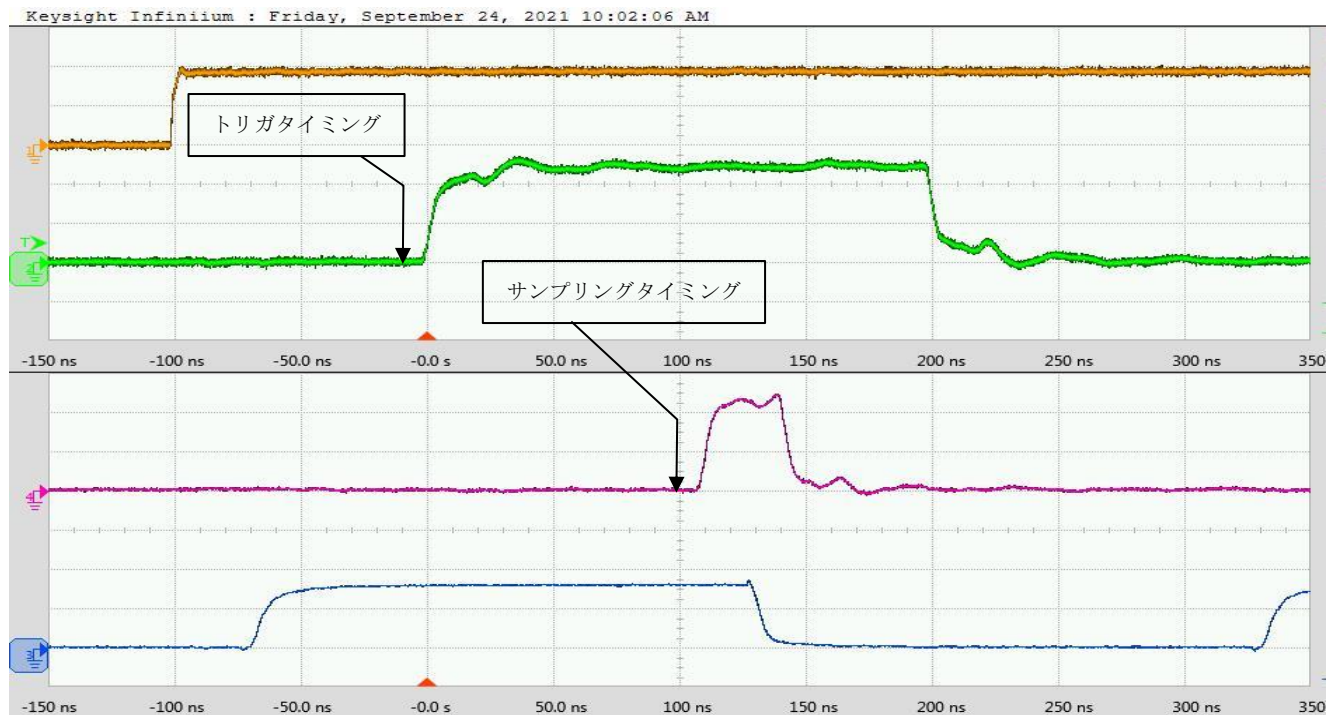
(2) Continuous モードと同じ結果と判断する。

(4-1) Internal 20MHz, Div8, Finite, Start Clock Delay 2ticks, Sample Clock Rising

ai/StartTrigger : SampleClockTimebase の同期ジッタ(最大 50ns)

処理時間 90ns+ 表示上遅延 10ns

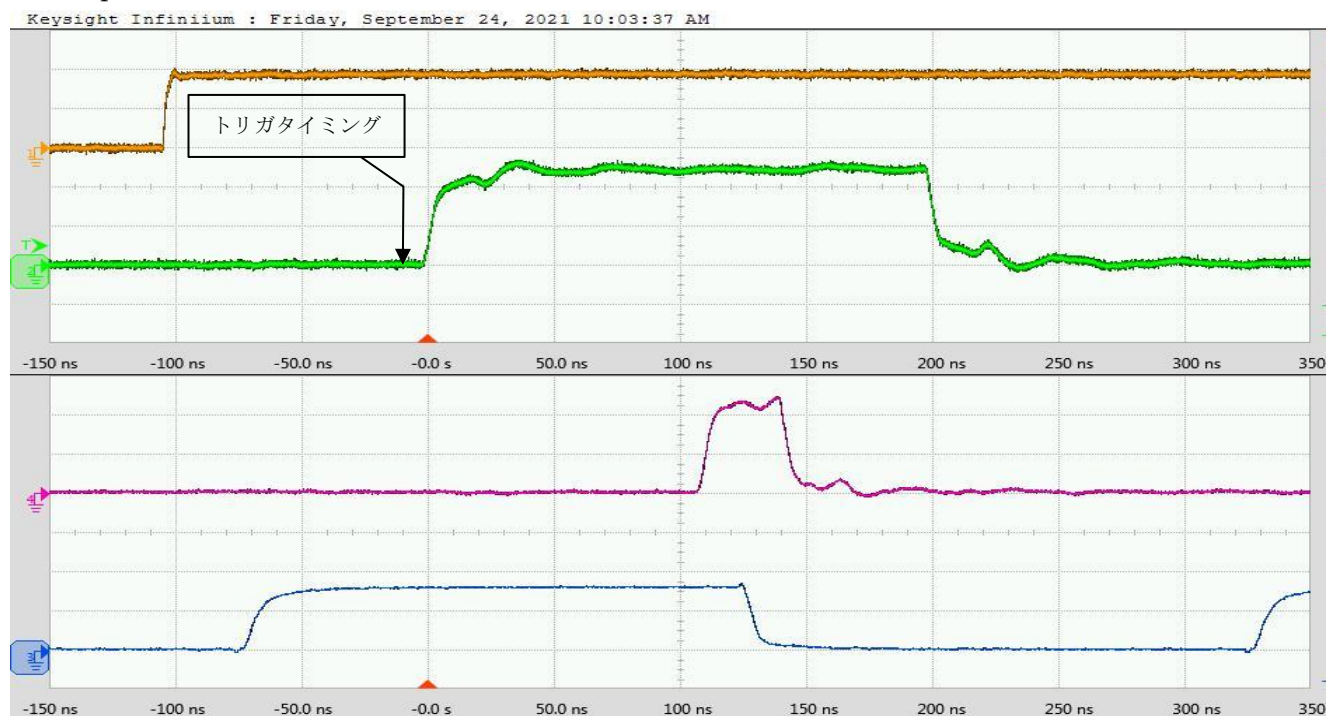
ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。 110ns (2ticks+10ns)



(4-2) (4-1) Sample Clock Rising → Falling

ai/StartTrigger : (4-1)と同じである。

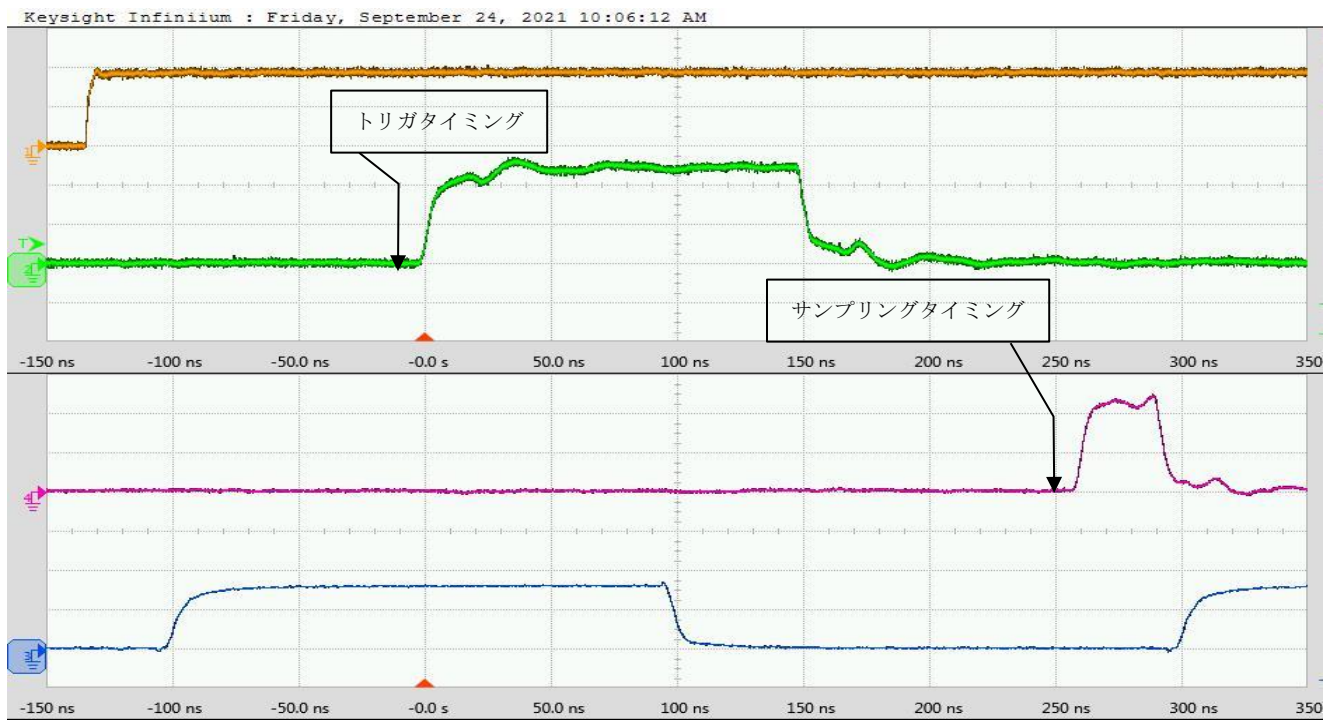
ai/SampleClock : (4-1)と同じである。



(4-3) (4-1) Start Clock Delay 2ticks → 5ticks

ai/StartTrigger : (4-1)と同じである。

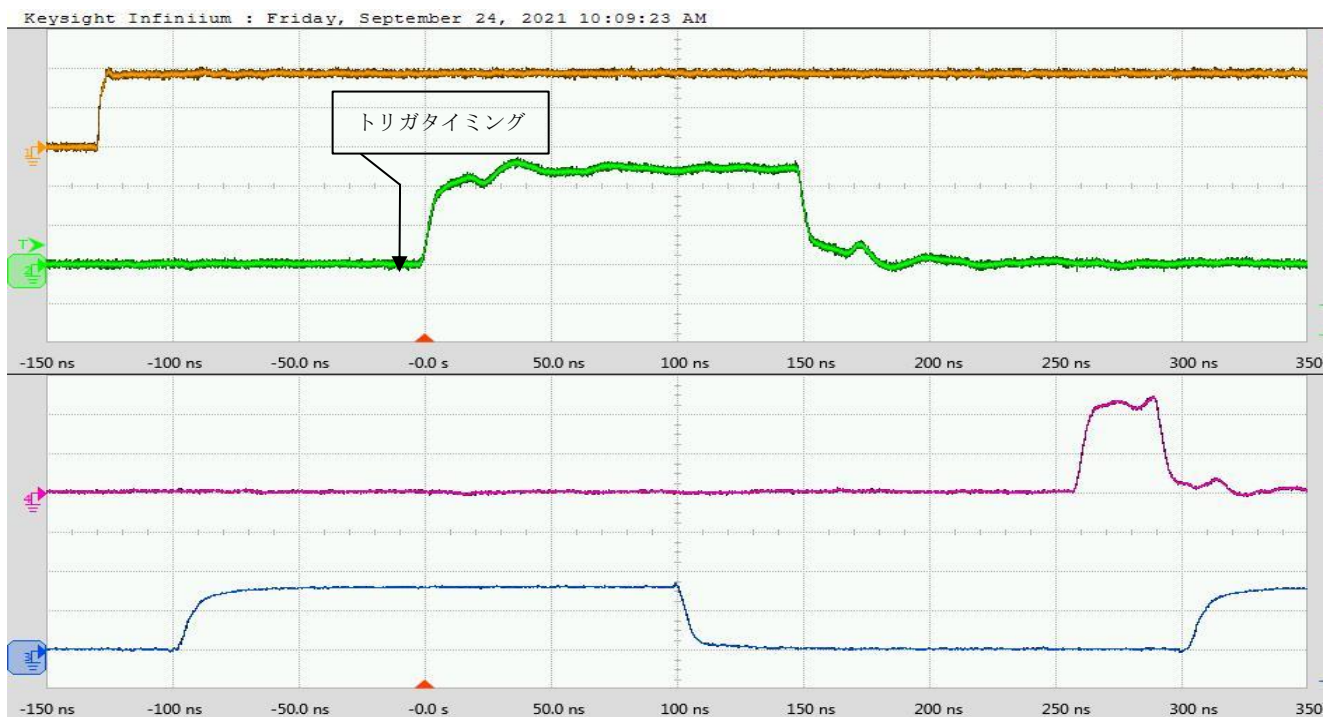
ai/SampleClock : ai/StartTrigger から Start Clock Delay+処理時間 10ns のずれ。260ns (5ticks+10ns)



(4-4) (4-3) Sample Clock Rising → Falling

ai/StartTrigger : (4-1)と同じである。

ai/SampleClock : (4-3)と同じである。

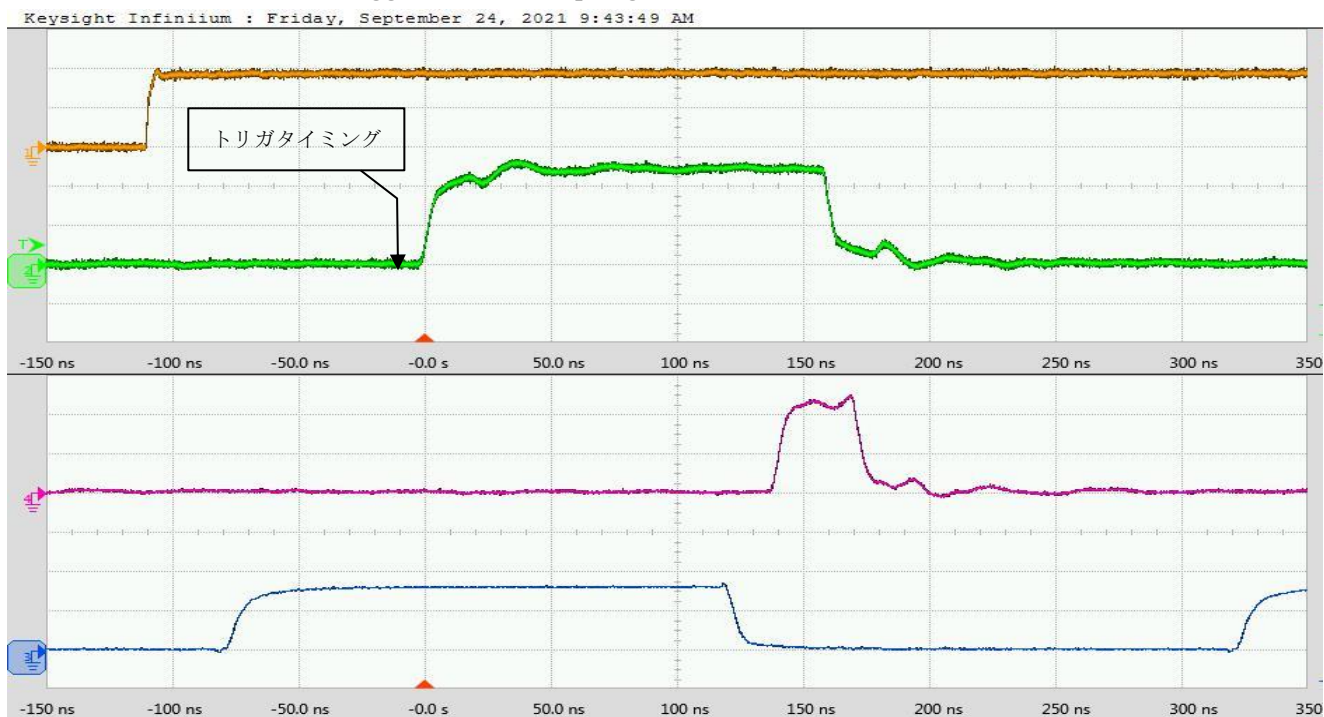


(5) Internal 100MHz, Div40, Finite PreSamples 5 : (3)のプリサンプルモード
SampleClockTimebaseActiveEdge=Rising である。

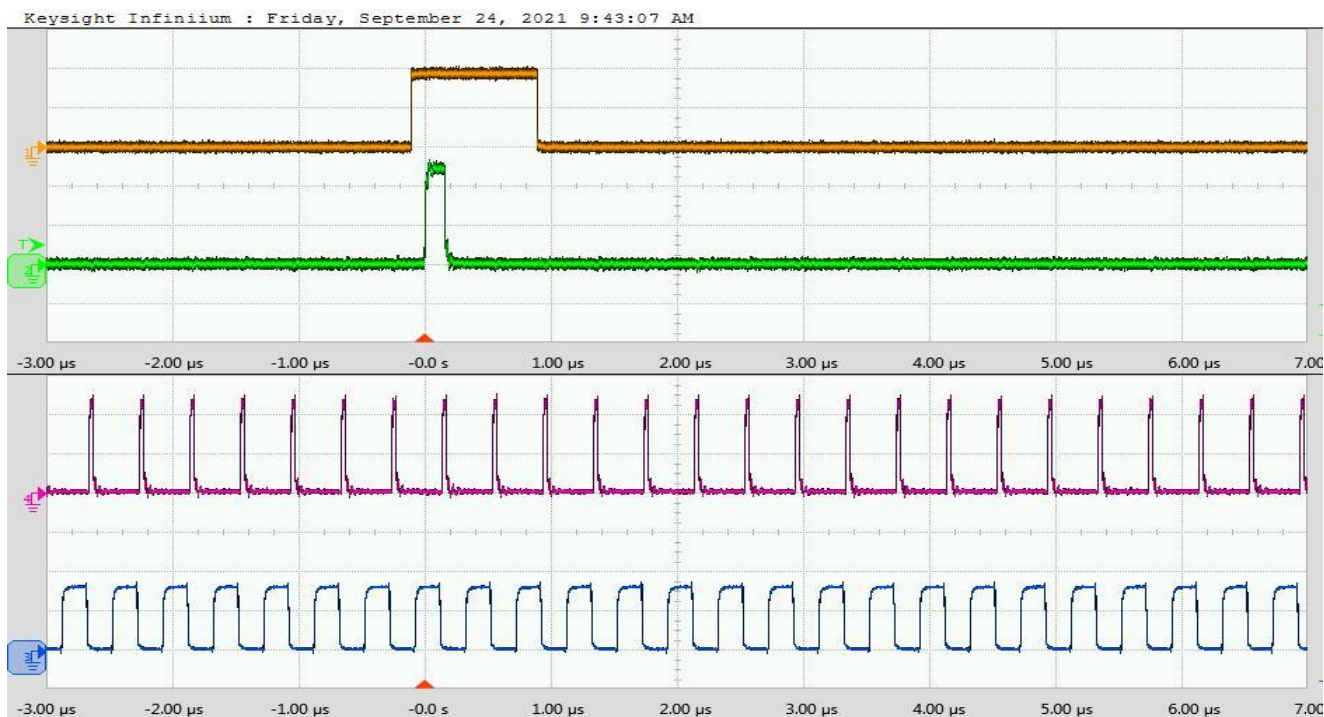
(5-1) Internal 100MHz, Div40 Finite PreSamples 5, (Sample Clock Delay 2), Sample Clock Rising
ai/StartTrigger : (3-1)と同じである。

ai/SampleClock : SampleClock の開始タイミングがトリガに依存せず、不定である。

ai/StartTrigger より、Sampling Rate の1クロック分のジッタが発生する。



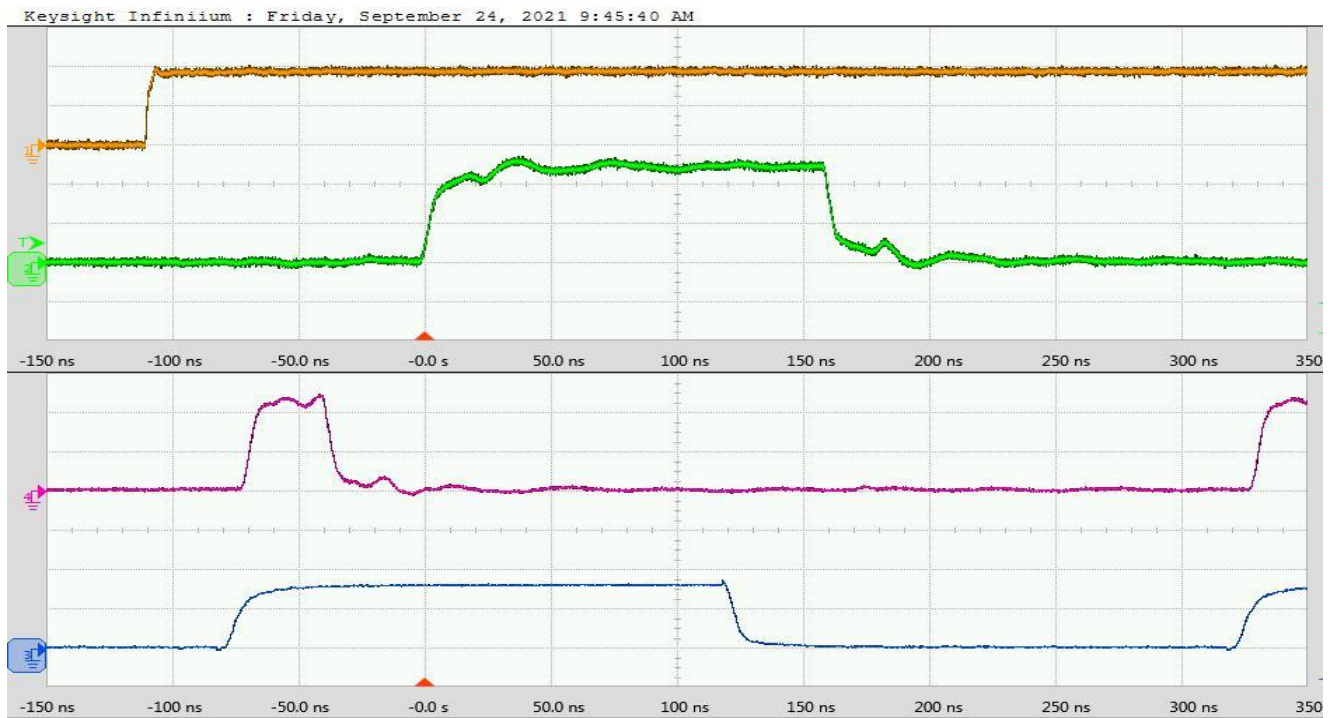
以下は全体図である。



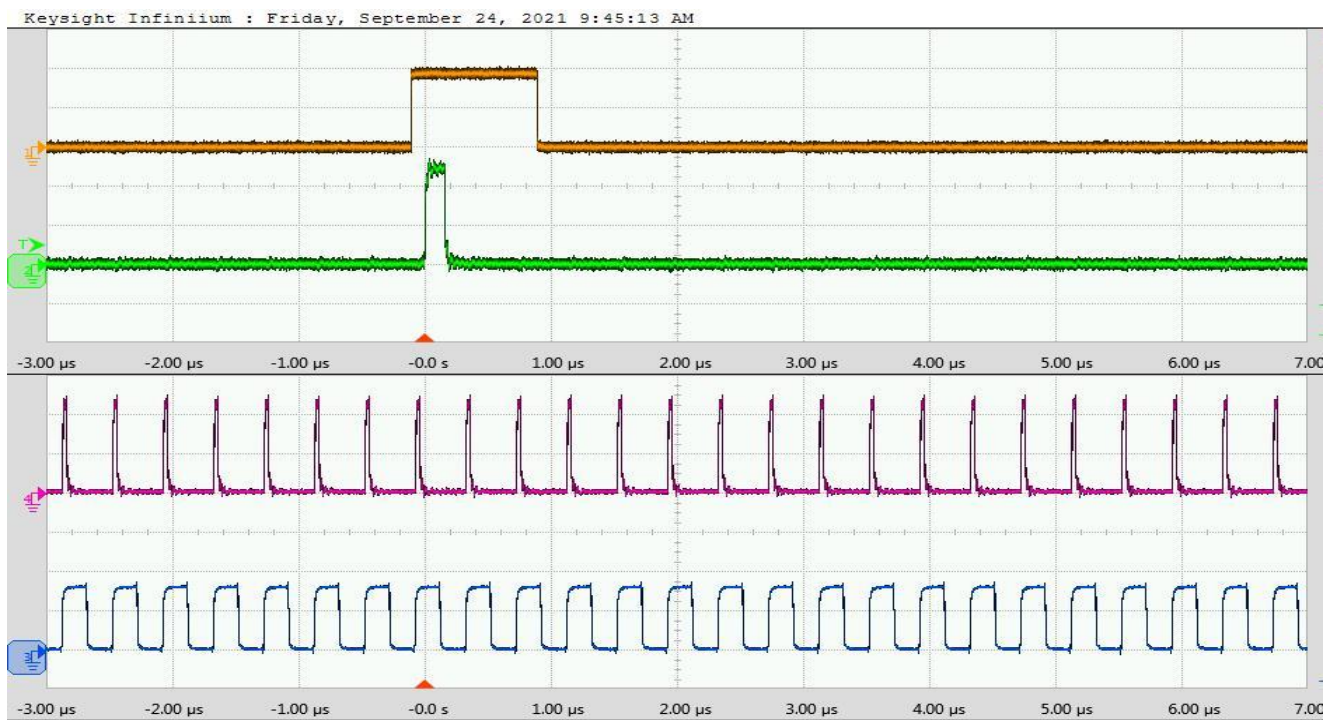
(5-2) (5-1) Sample Clock Rising → Falling

ai/StartTrigger : (3-1)と同じである。

ai/SampleClock : (5-1)と同じである。



以下は全体図である。

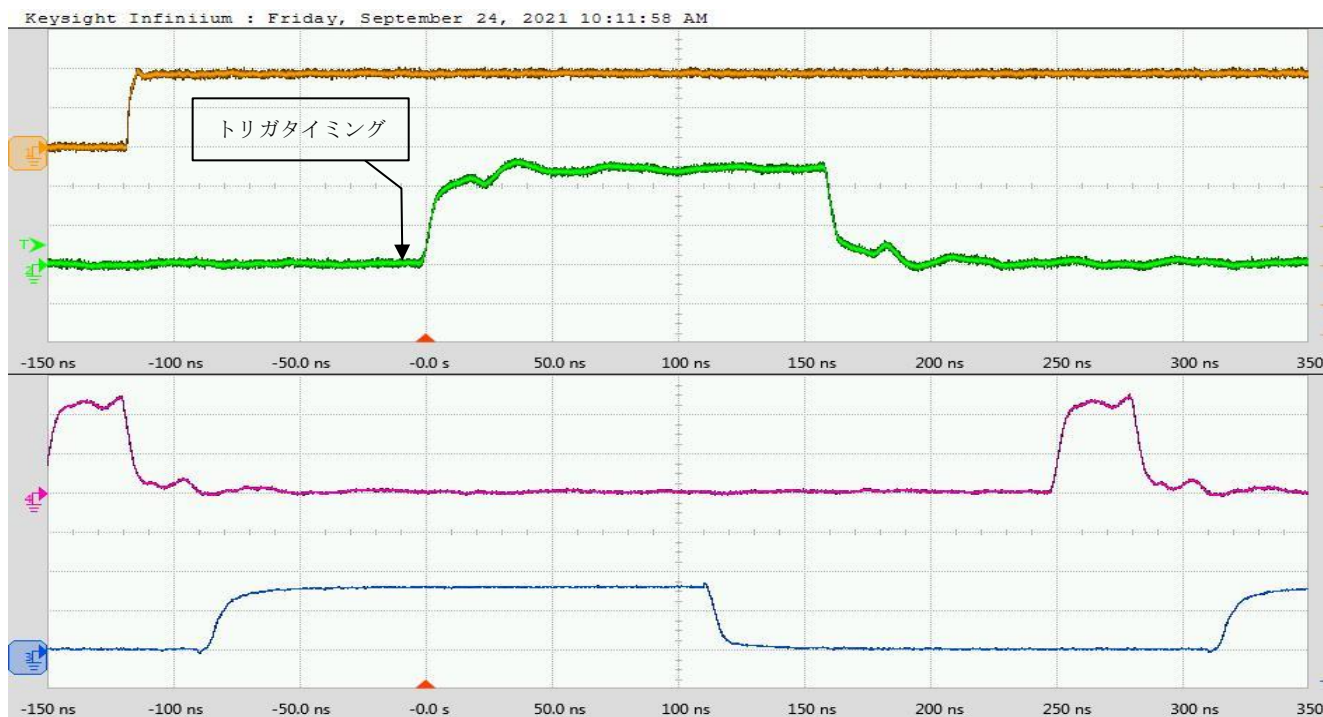


(6) Internal 20MHz, Div8, Finite PreSamples 5 : (4)のプリサンプルモード
SampleClockTimebaseActiveEdge=Rising である。

(6-1) Internal 20MHz, Div8 Finite PreSamples 5, (Sample Clock Delay 2), Sample Clock Rising
ai/StartTrigger : (4-1)と同じである。

ai/SampleClock : SampleClock の開始タイミングがトリガに依存せず、不定である。

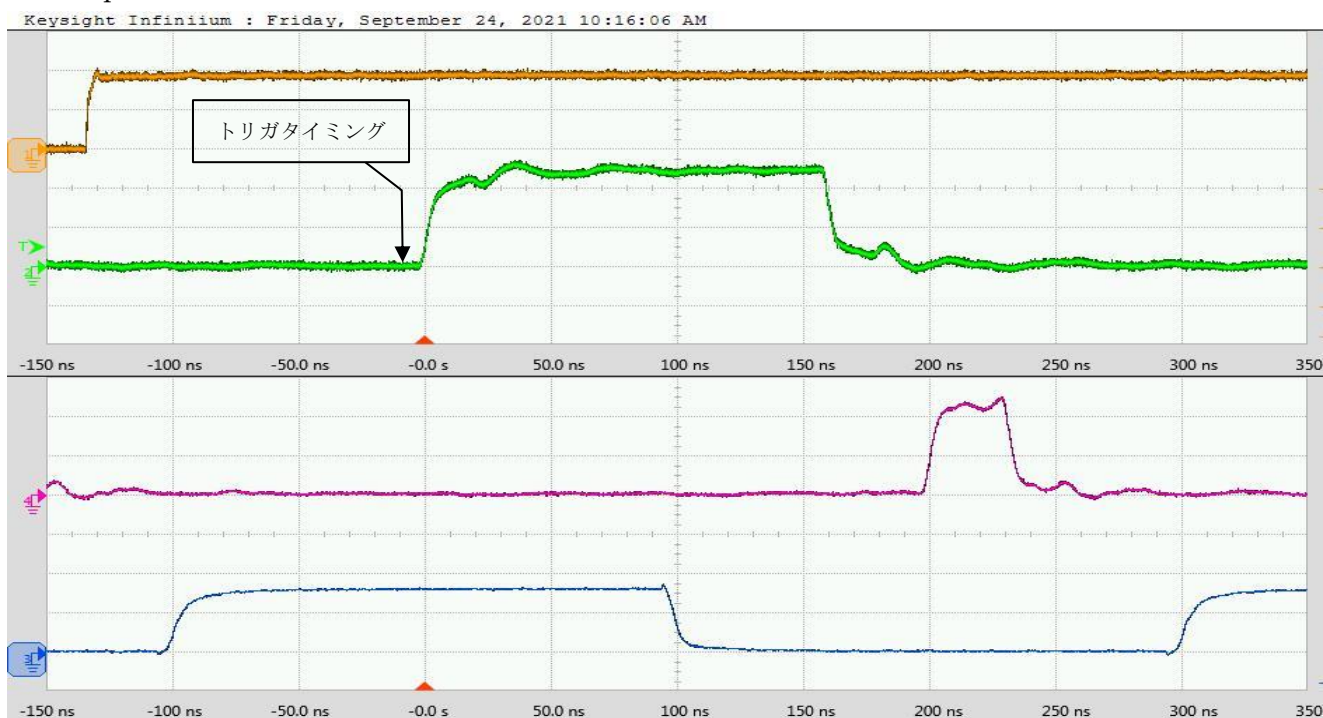
ai/StartTrigger より、Sampling Rate の1クロック分のジッタが発生する。



(6-2) (6-1) Sample Clock Rising → Falling

ai/StartTrigger : (4-1)と同じである。

ai/SampleClock : (6-1)と同じである。



(7) External 10MHz, Div4, Continuous

SampleClockTimebaseActiveEdge=Rising で動作する。

SampleClockTimebase に External 10MHz(デモジュレータのクロック)信号を接続する。

(7-1) External 10MHz, Div4, Continuous, Start Clock Delay 2ticks, Sample Clock Rising

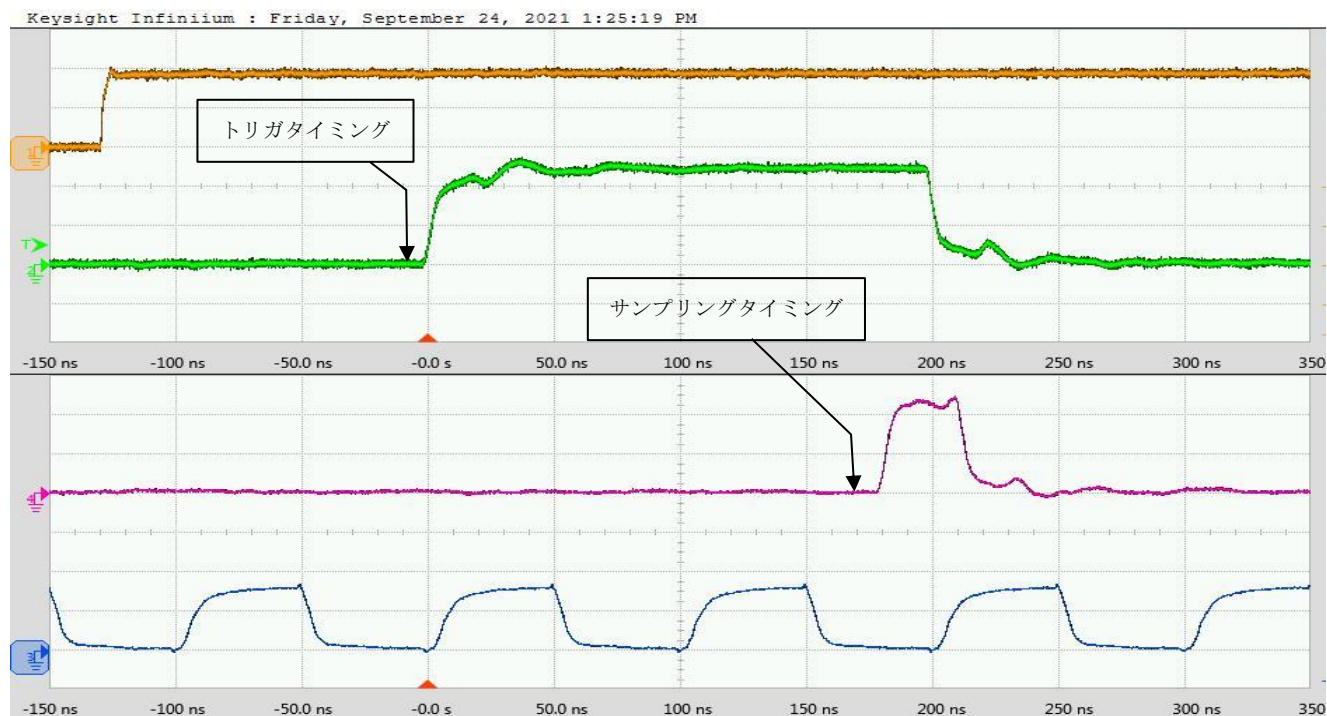
ai/StartTrigger : デモジュレータのトリガとクロックの遅延時間(20~30ns)

+ 処理時間 90ns+ 表示上遅延 10ns

ai/SampleClock : 第1外部クロック Rising (≒デモジュレータのトリガ)から

Start Clock Delay×外部クロック周期+処理時間 70ns+表示上遅延 10ns のずれ。

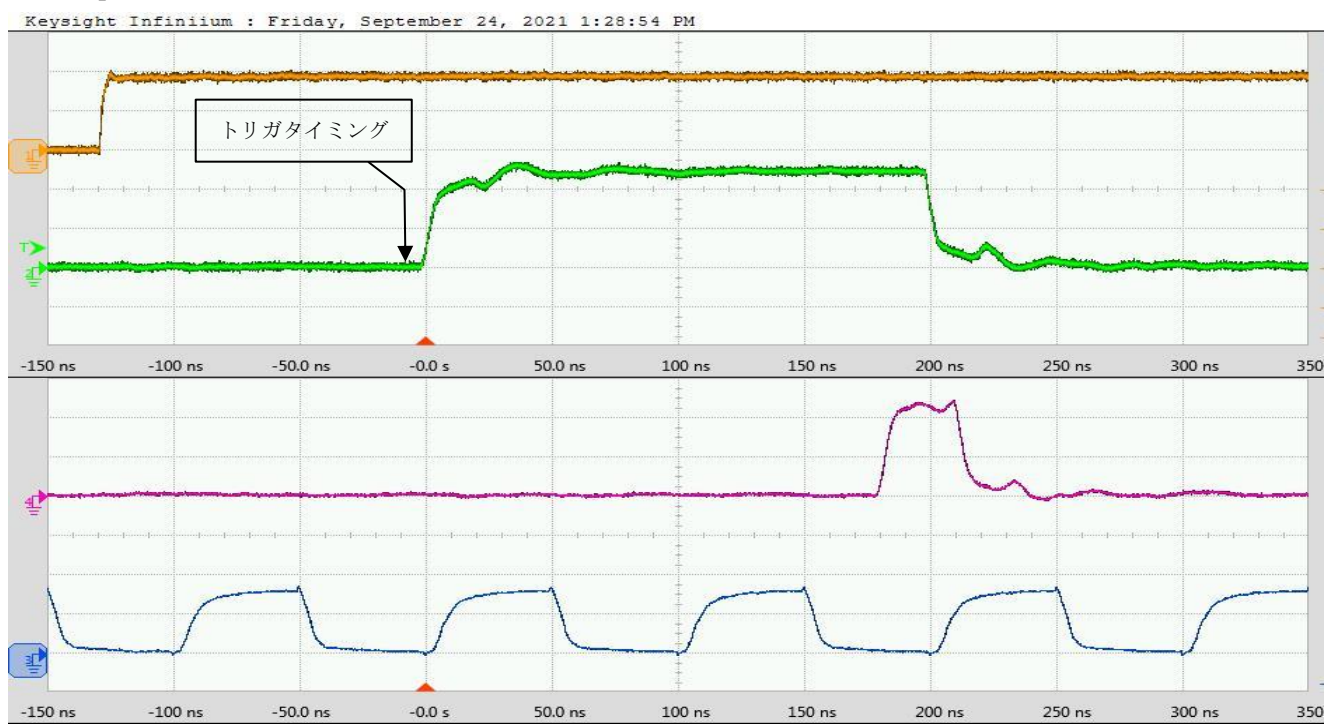
200ns(2ticks)+80ns



(7-2) (7-1) Sample Clock Rising → Falling

ai/StartTrigger : (7-1)と同じである。

ai/SampleClock : (7-1)と同じである。



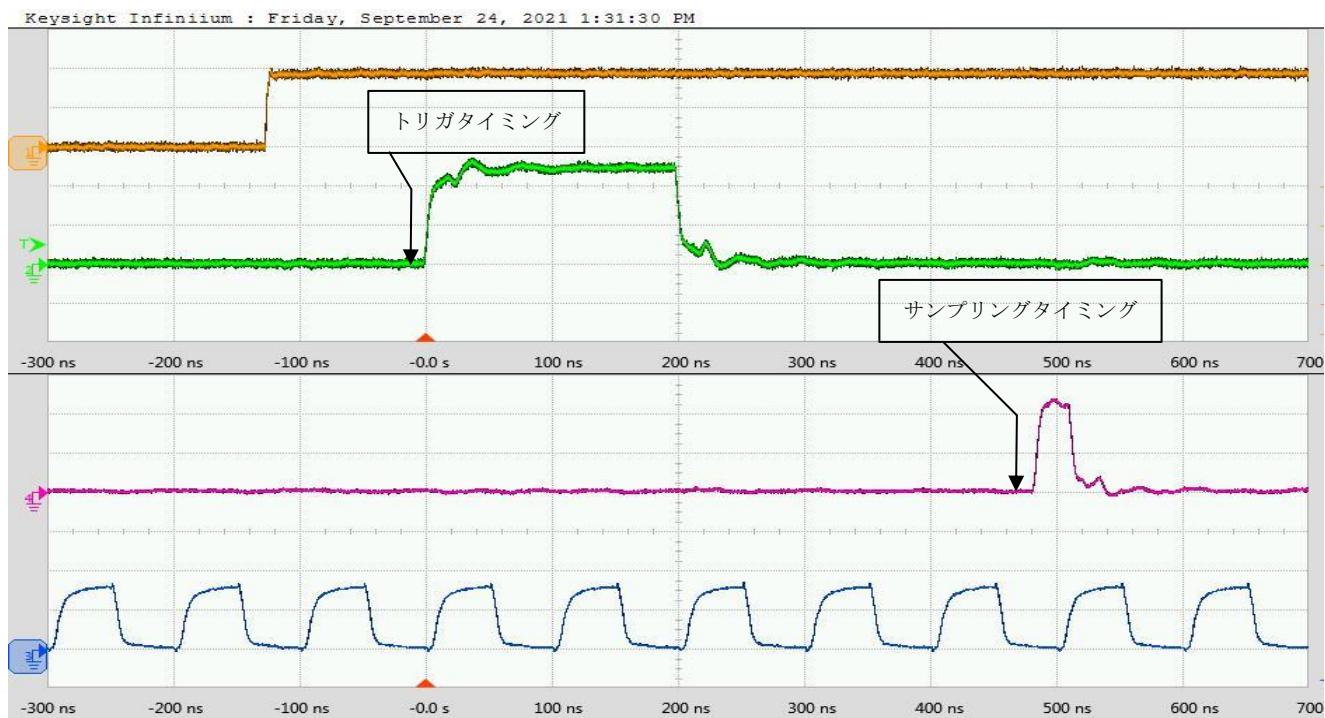
(7-3) (7-1) Start Clock Delay 2ticks → 5ticks

ai/StartTrigger : (7-1) と同じである。

ai/SampleClock : 第1外部クロック Rising (≒デモジェレータのトリガ)から

Start Clock Delay×外部クロック周期+処理時間 70ns+表示上遅延 10ns のずれ。

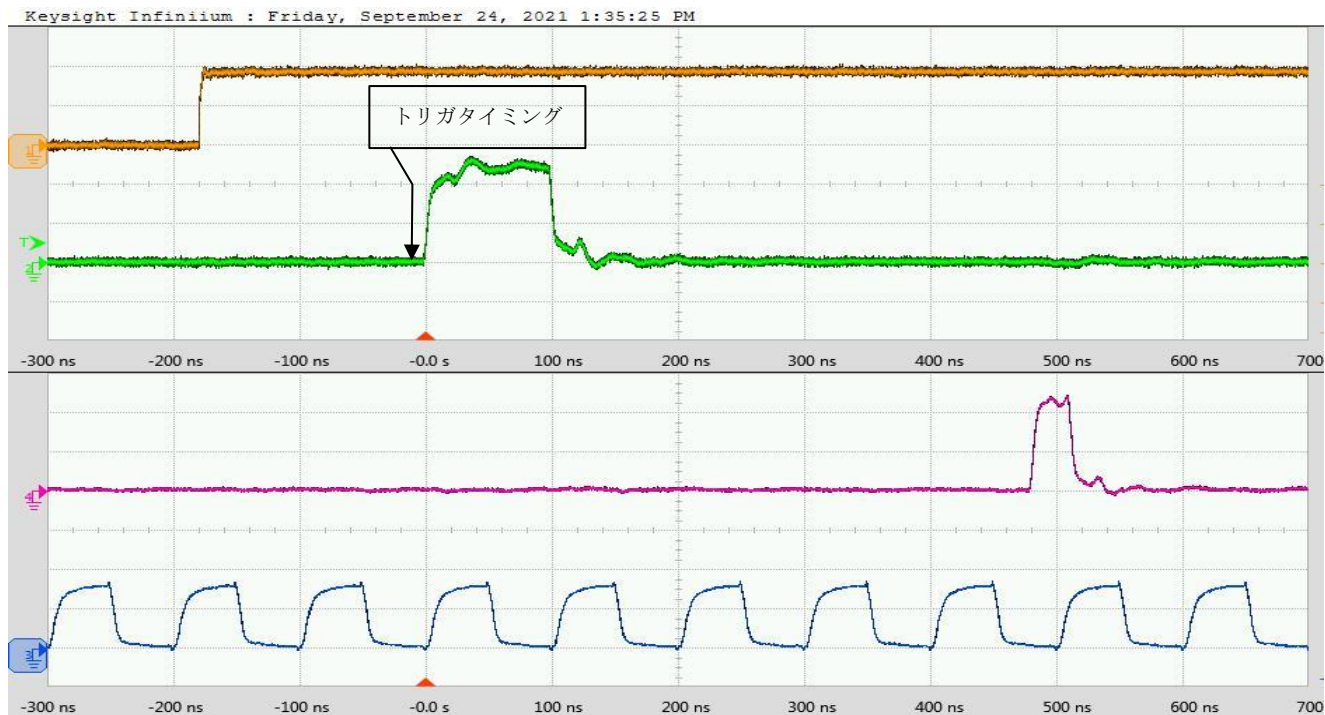
500ns(5ticks)+80ns



(7-4) (7-3) Sample Clock Rising → Falling

ai/StartTrigger : (7-1) と同じである。

ai/SampleClock : (7-3) と同じである。



(8) External 10MHz, Div4, Finite : (7)の Finite モード

(7) Continuous モードと同じ結果と判断する。

(8-1) External 10MHz, Div4, Finite, Start Clock Delay 2ticks, Sample Clock Rising

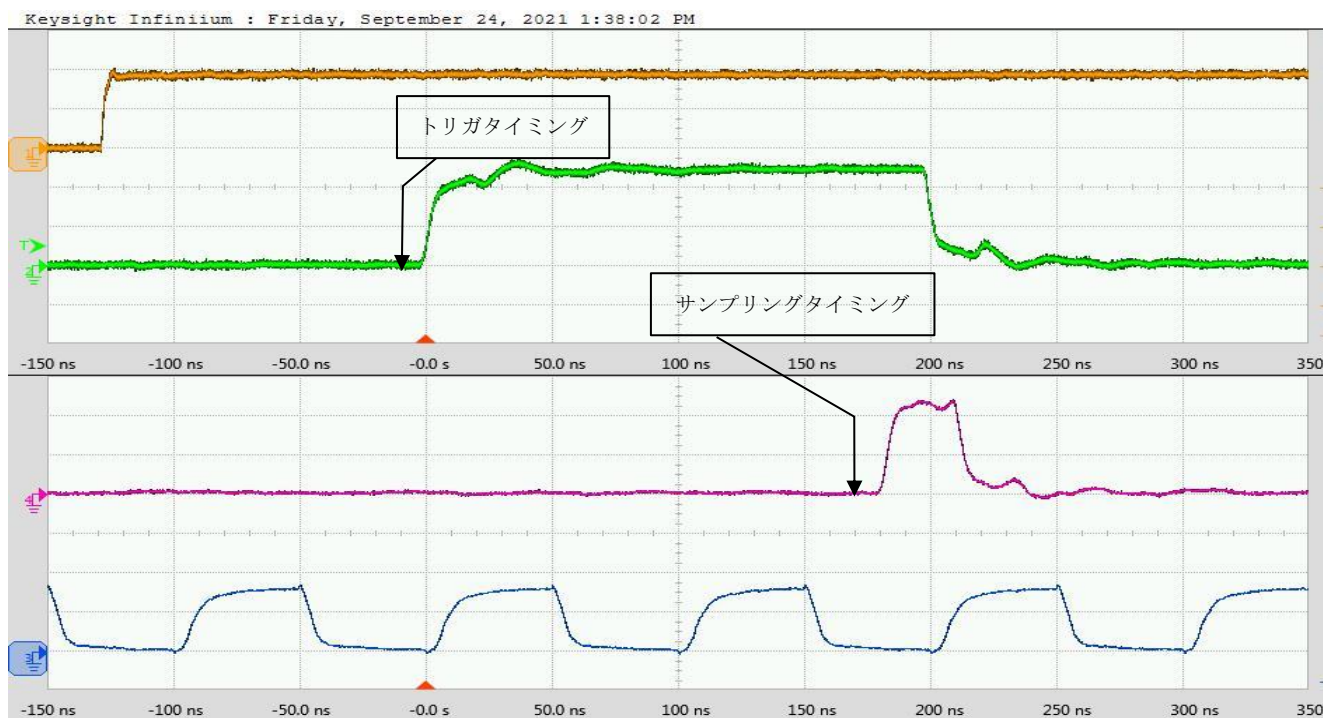
ai/StartTrigger : デモジュレータのトリガとクロックの遅延時間(20~30ns)

+ 処理時間 90ns+ 表示上遅延 10ns

ai/SampleClock : 第1外部クロック Rising (≒デモジュレータのトリガ)から

Start Clock Delay×外部クロック周期+処理時間 70ns+表示上遅延 10ns のずれ。

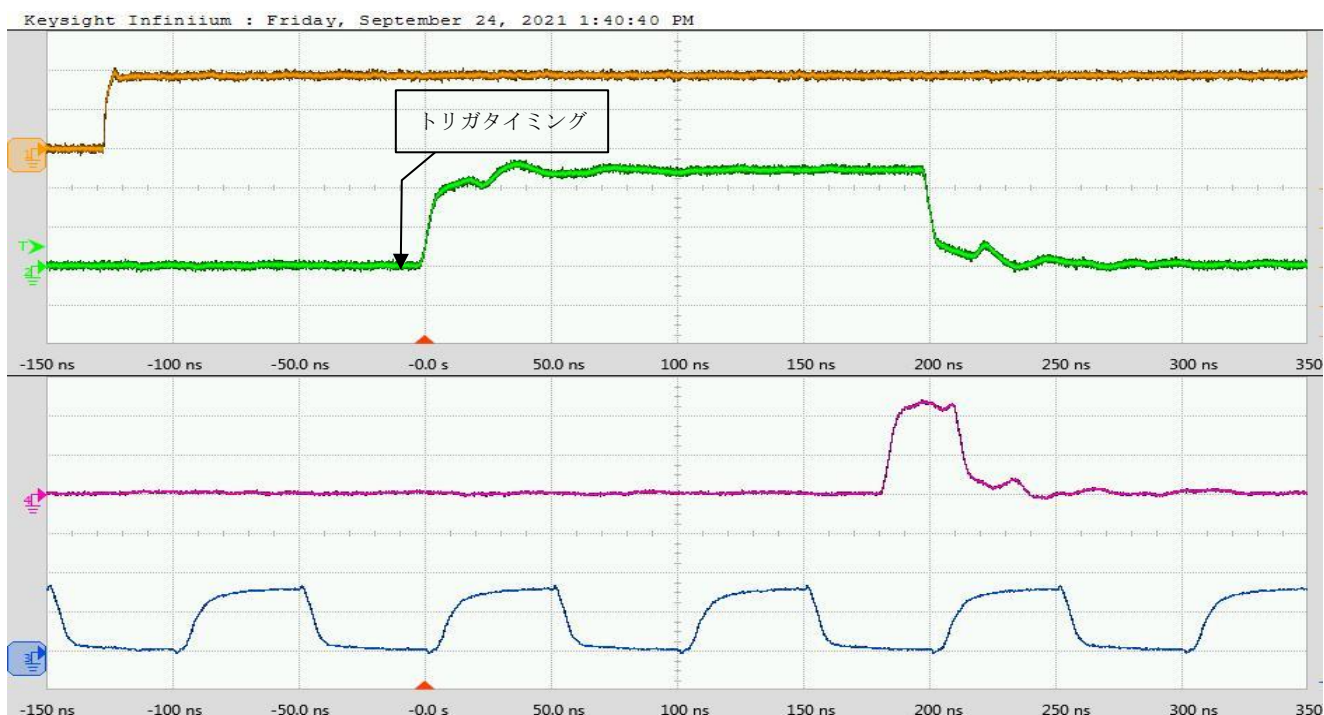
200ns(2ticks)+80ns



(8-2) (8-1) Sample Clock Rising → Falling

ai/StartTrigger : (8-1) と同じである。

ai/SampleClock : (8-1) と同じである。



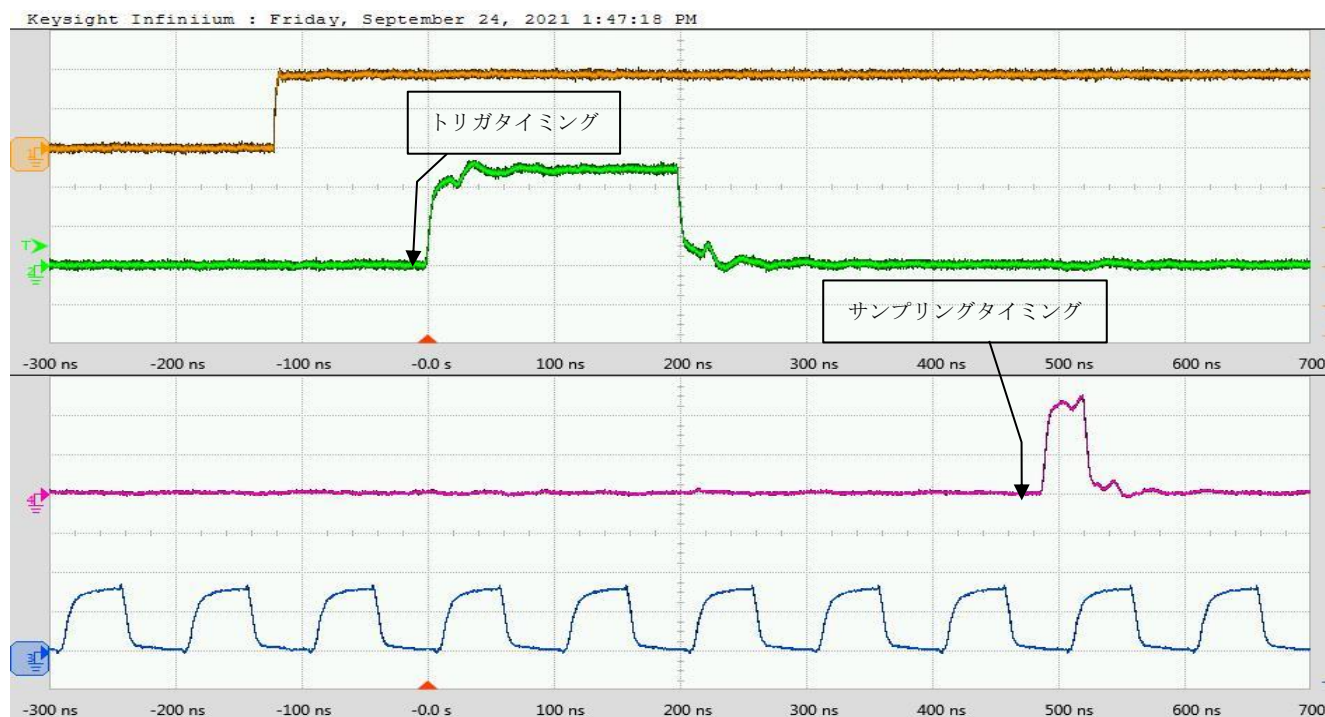
(8-3) (8-1) Start Clock Delay 2ticks → 5ticks

ai/StartTrigger : (8-1) と同じである。

ai/SampleClock : 第1外部クロック Rising (≒デモジェレータのトリガ)から

Start Clock Delay×外部クロック周期+処理時間 70ns+表示上遅延 10ns のずれ。

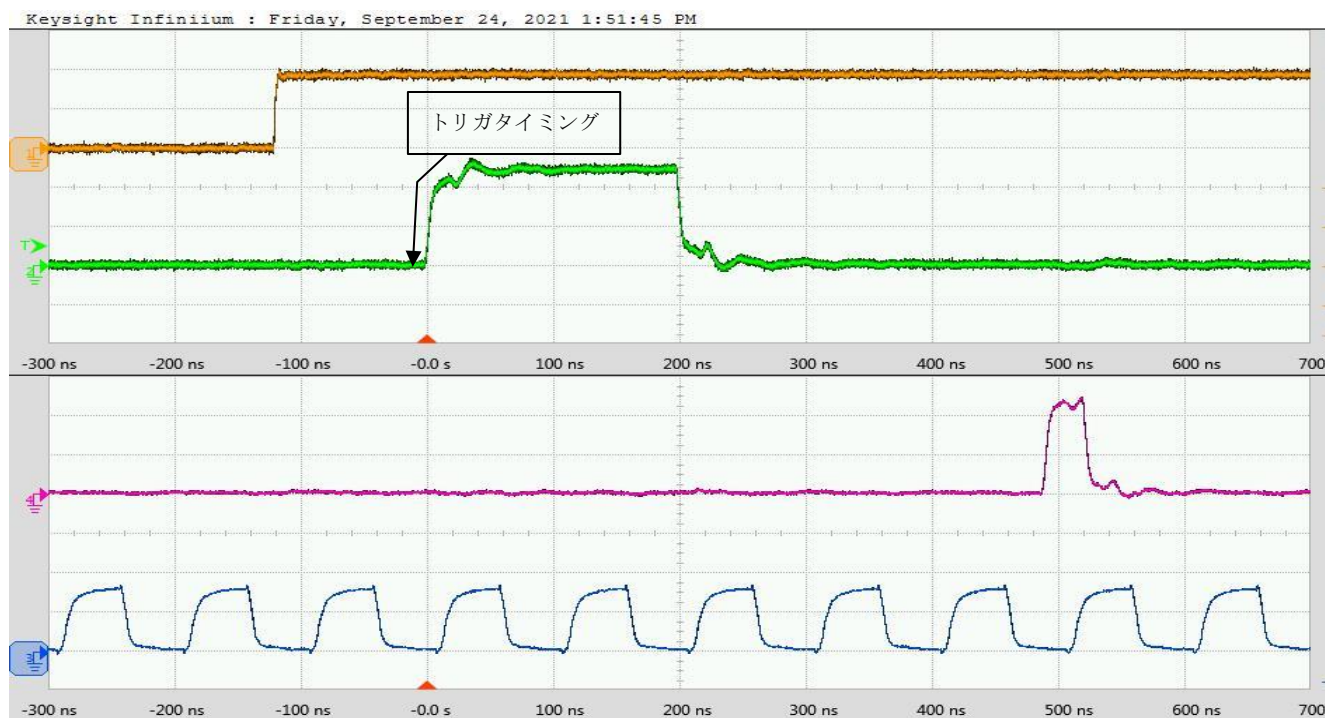
500ns(5ticks)+80ns



(8-4) (8-3) Sample Clock Rising → Falling

ai/StartTrigger : (8-1) と同じである。

ai/SampleClock : (8-1) と同じである。



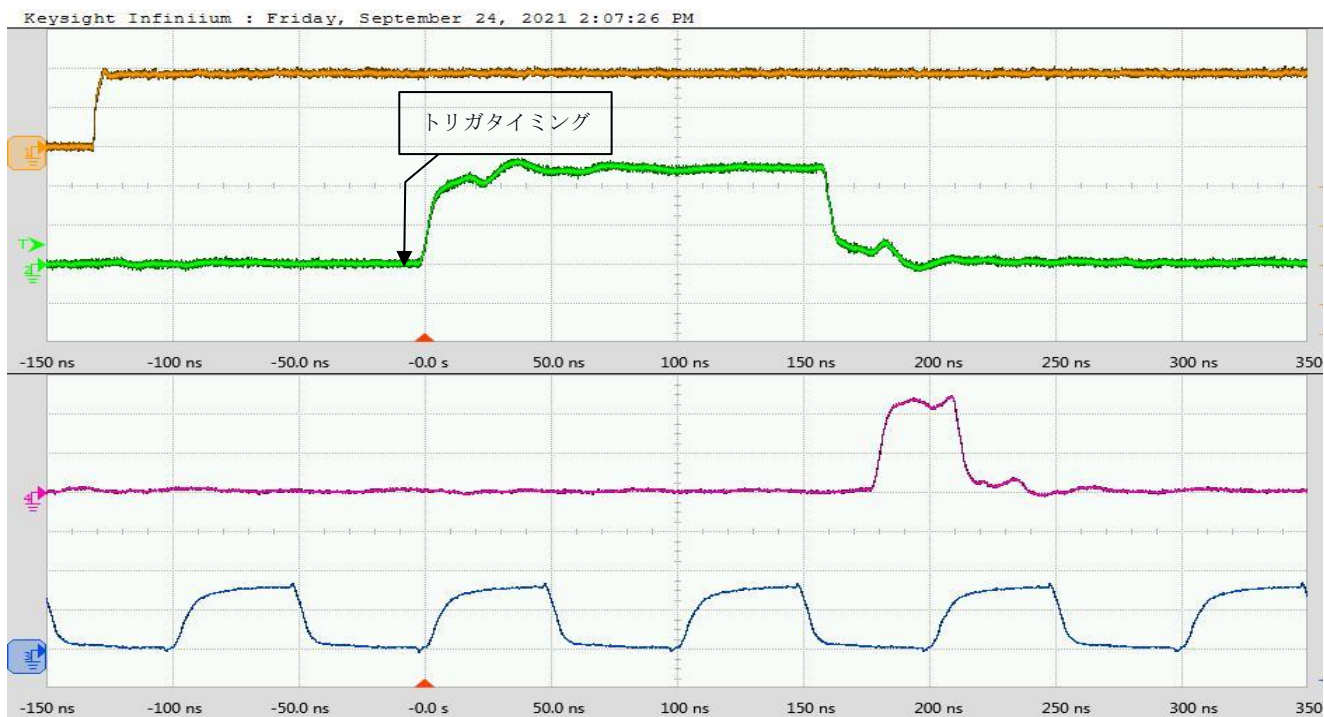
(9) External 10MHz, Div4, Finite PreSamples 5

ai/StartTrigger : (8-1)と同じである。

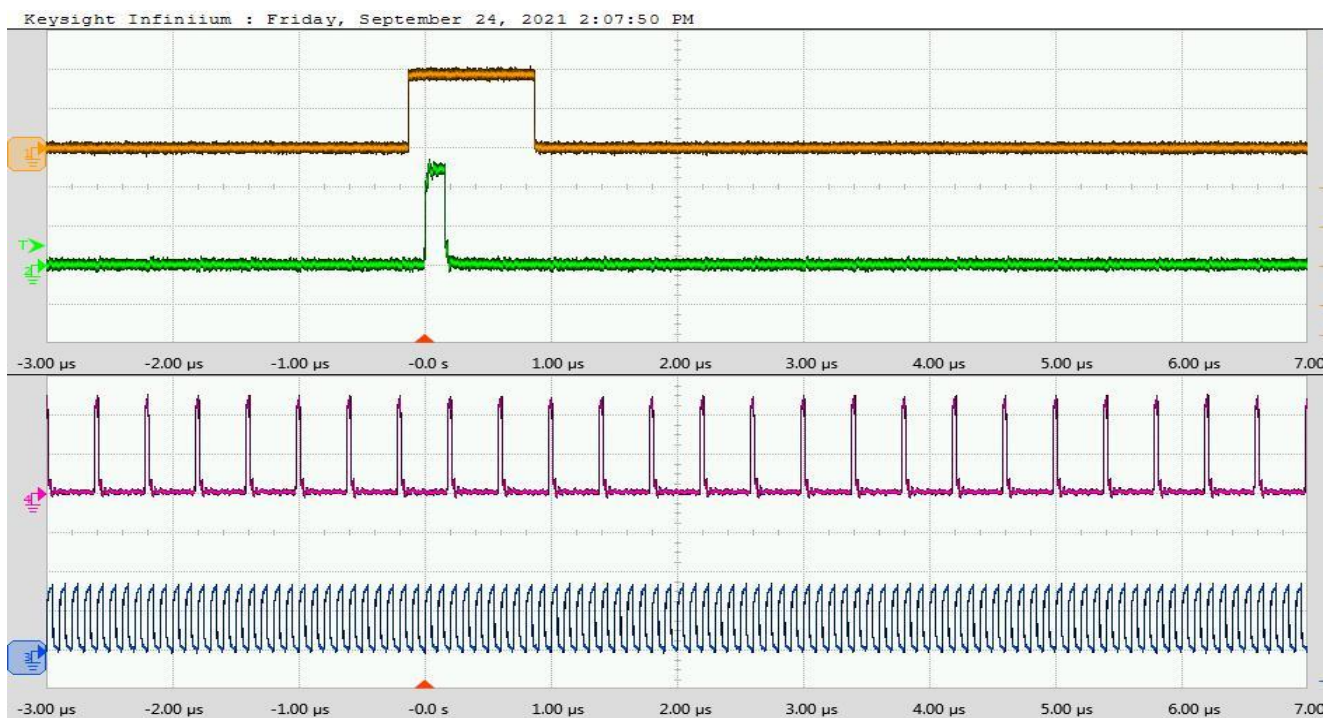
ai/SampleClock : SampleClock の開始タイミングがトリガに依存せず、不定である。

ai/StartTrigger より、Sampling Rate の1クロック分のジッタが発生する。

(9-1) External 10MHz, Div4, Finite PreSamples 5, (Sample Clock Delay 2), Sample Clock Rising

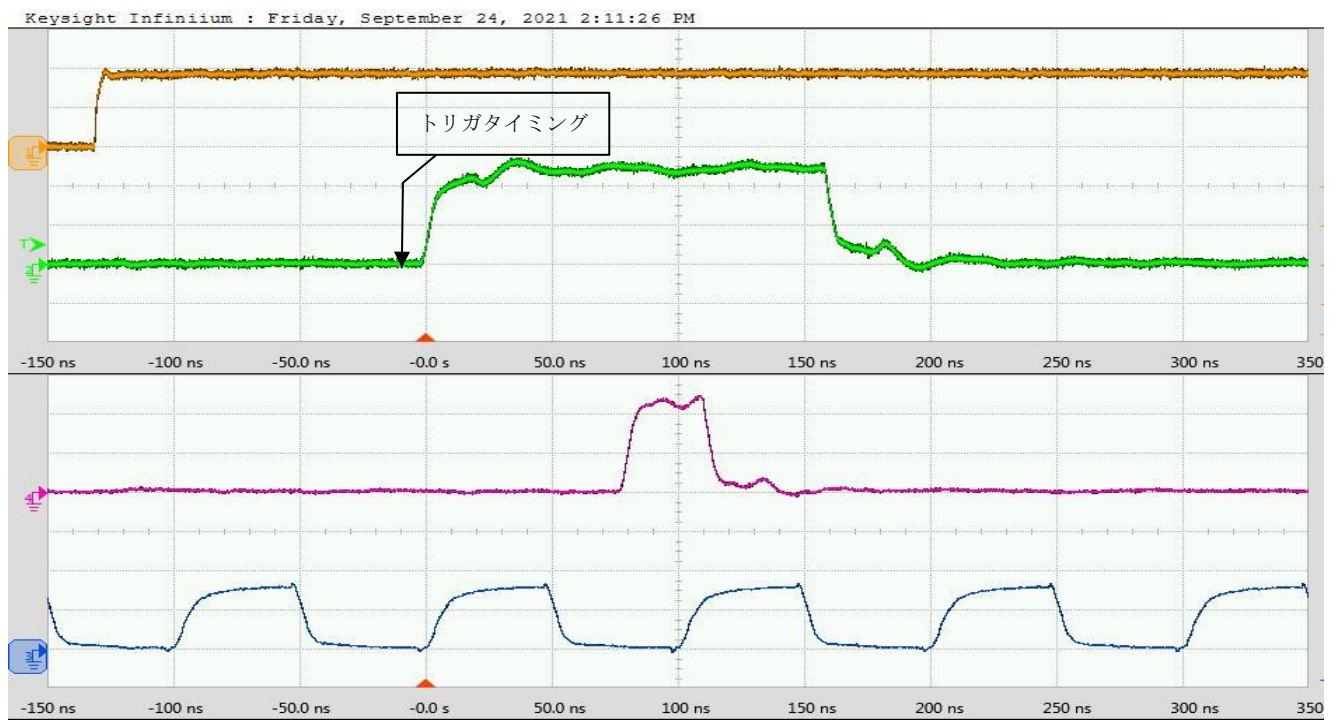


以下は全体図である。

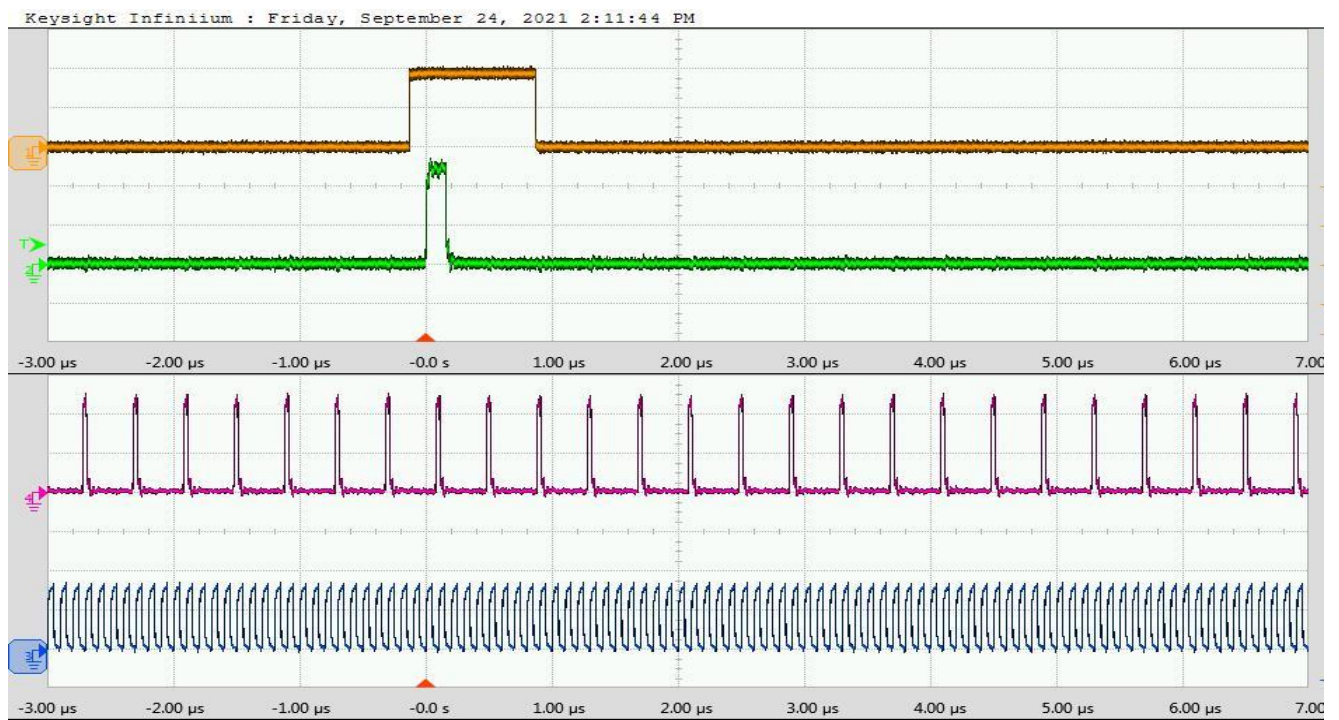


(9-2) (9-1) Sample Clock Rising → Falling

Rising と同じである。



以下は全体図である。



(10) External 2.5MHz, Div1, Continuous

SampleClockTimebaseは Internal 100MHz 信号が接続されている。

SampleClockTimebaseActiveEdge=Rising(設定値)で動作する。

SampleClockに External 2.5MHz(デモジュレータのクロック)信号を接続する。

(10-1) External 2.5MHz, Div1, Continuous, (Start Clock Delay 2ticks), Sample Clock Rising

ai/StartTrigger : 処理時間 90ns+10ns(表示上遅延)+ SampleClockTimebase のジッタ 10ns

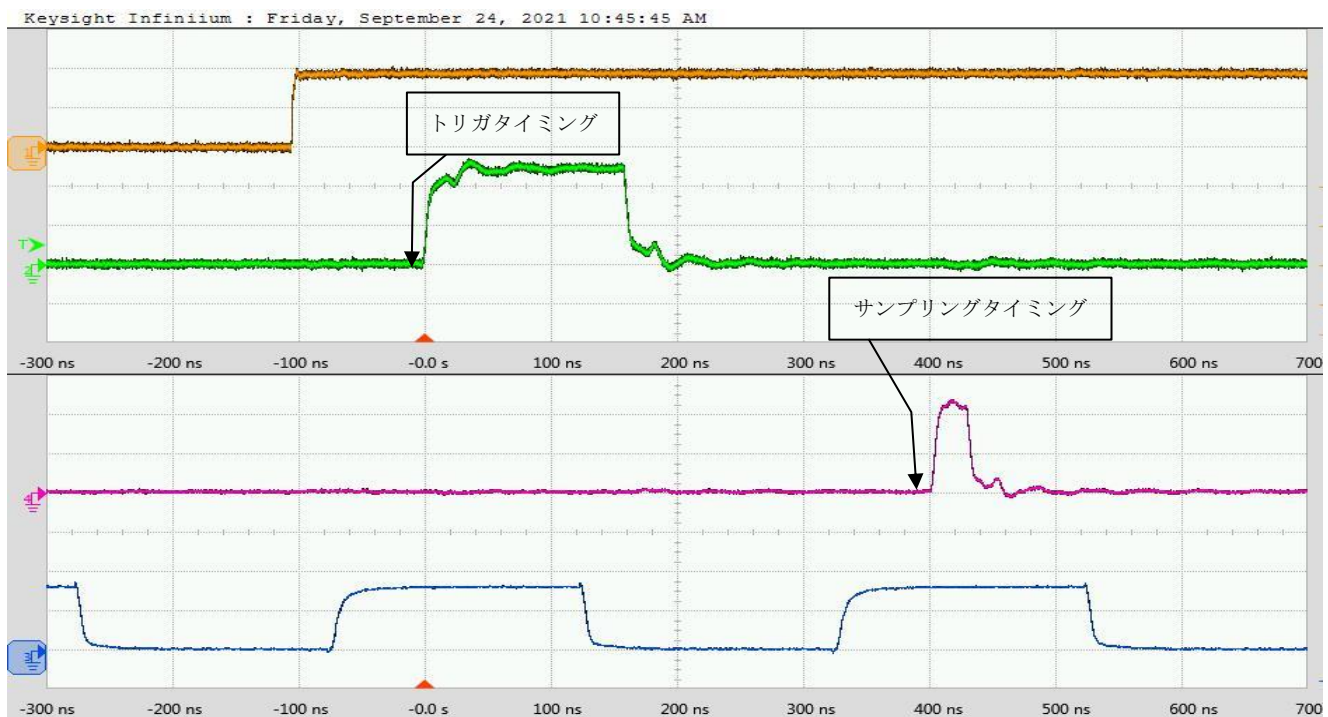
第1 外部クロック Rising から処理時間 70ns+10ns(表示上遅延)

ai/SampleClock : 第1 外部クロック Rising (=デモジュレータのトリガ)から

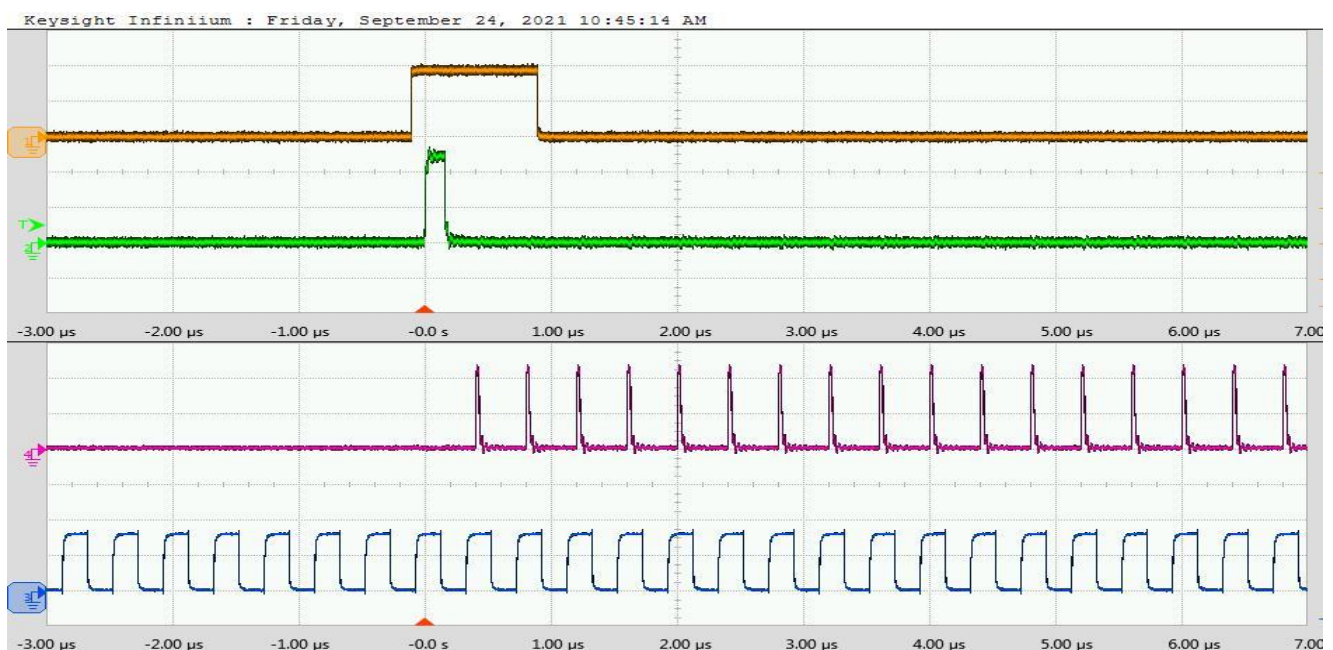
外部クロック周期+処理時間 70ns+表示上遅延 10ns のずれ 2us+80ns

第2 外部クロック Rising から処理時間 70ns+10ns(表示上遅延)

(ai/StartTrigger 以降の第1 外部クロック)



以下、全体図

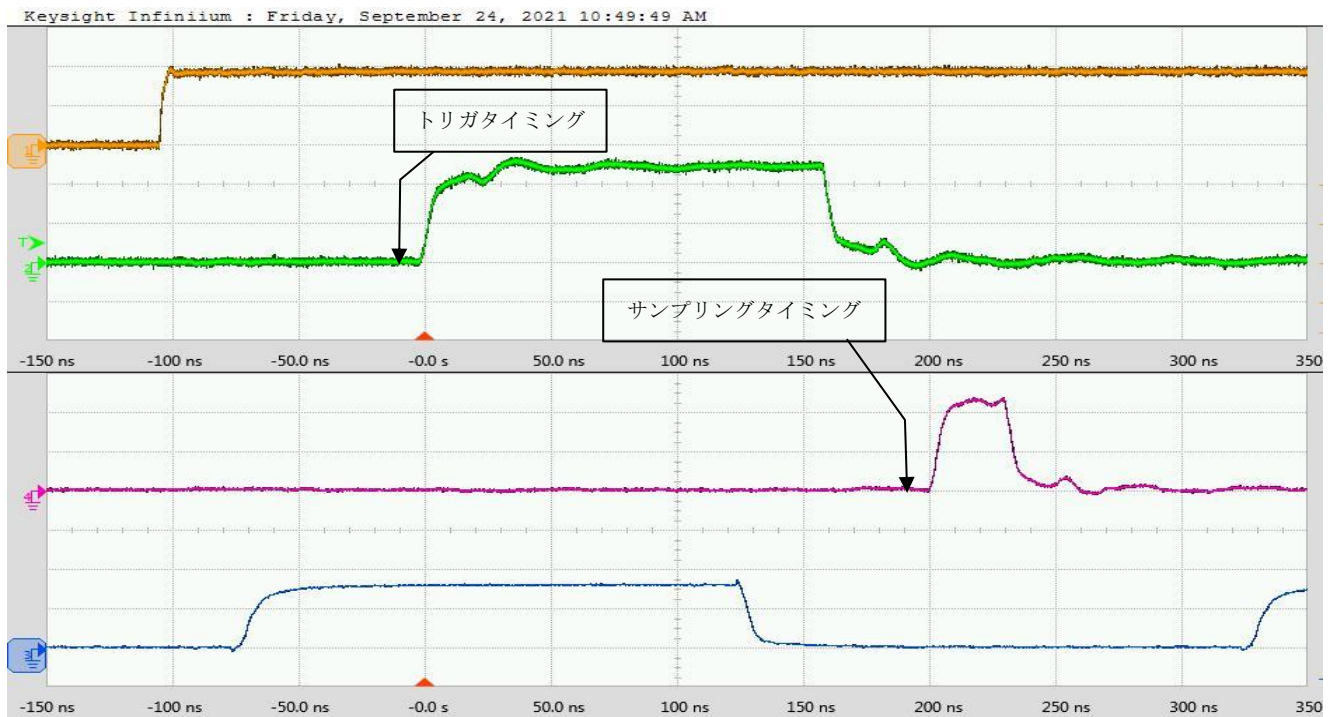


(10-2) (10-1) Sample Clock Rising → Falling

ai/StartTrigger : (11-1)と同じである。

ai/SampleClock : ai/StartTrigger 以降の第1 Falling から処理時間 70ns+10ns(表示上遅延)

第1 外部クロック Falling から処理時間 70ns+10ns(表示上遅延)



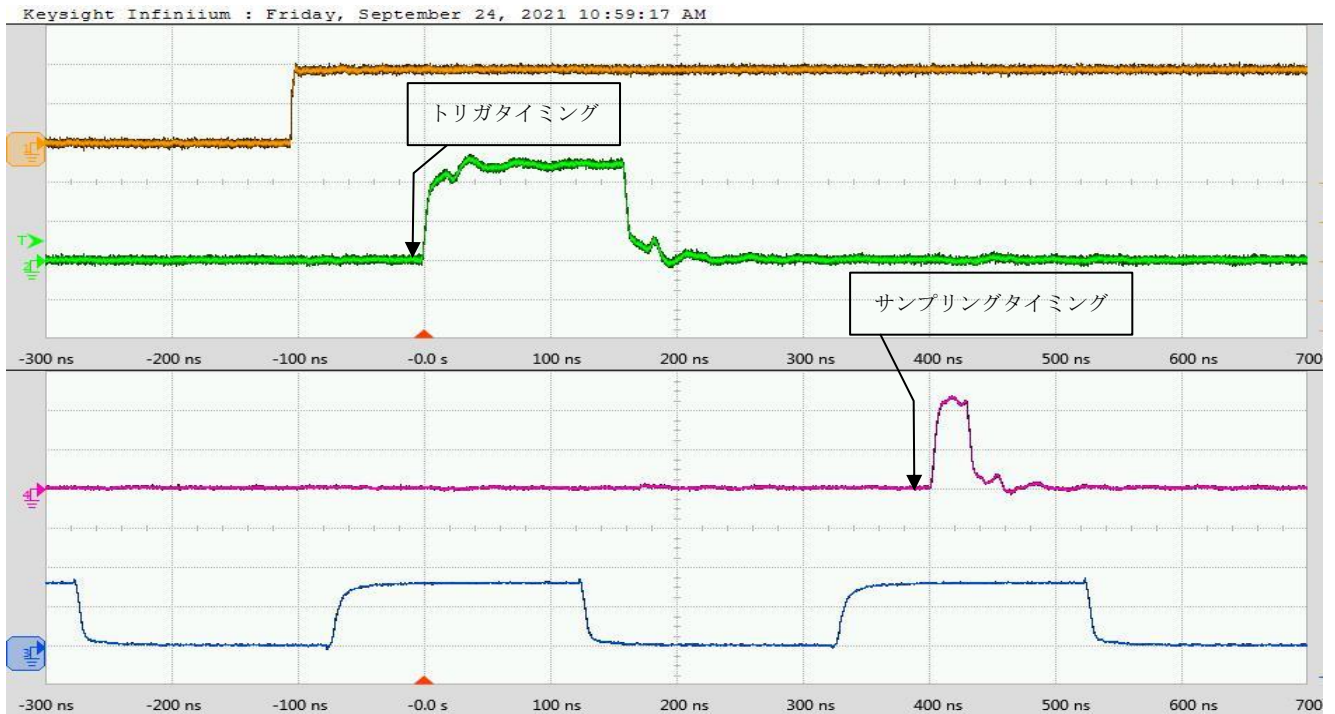
(11) External 2.5MHz, Div1, Finite

(10)と同じである。

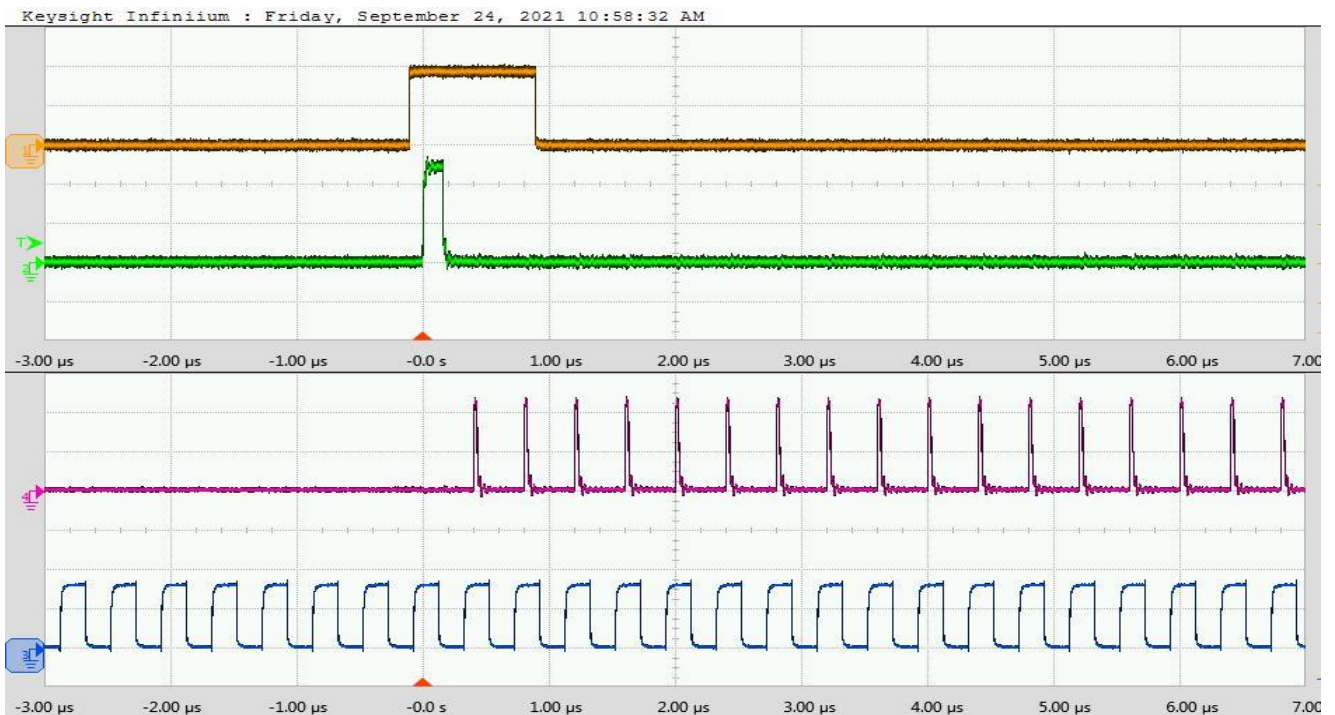
(11-1) External 2.5MHz, Div1, Finite, (Start Clock Delay 2ticks), Sample Clock Rising

ai/StartTrigger : (10-1)と同じである。

ai/SampleClock : (10-1)と同じである。



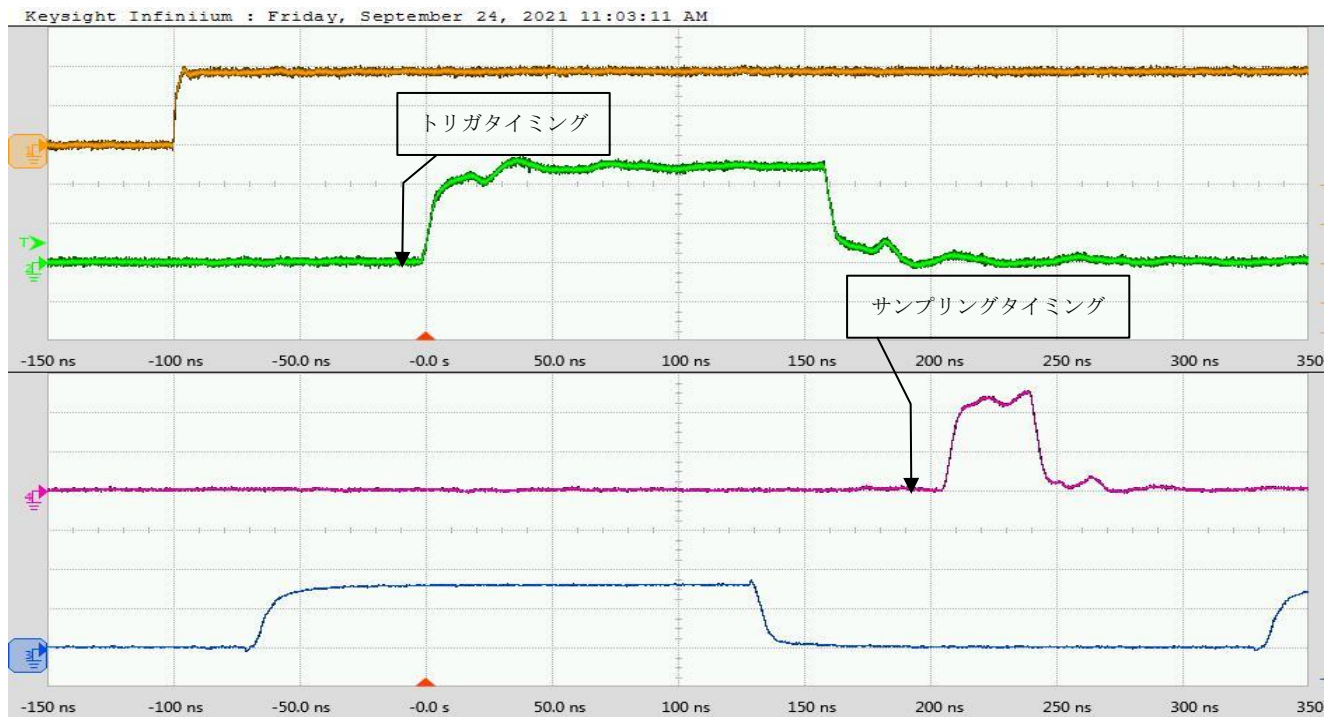
以下は全体図である。



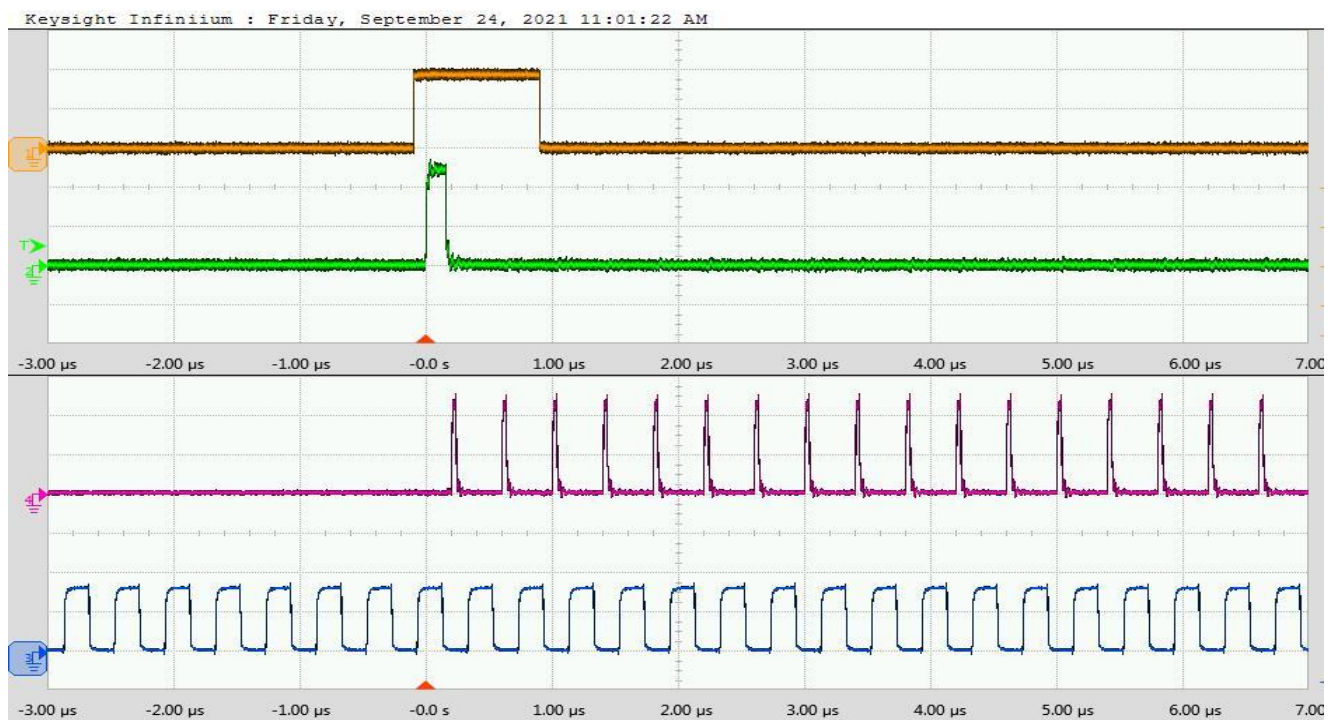
(11-2) (11-1) Sample Clock Rising → Falling

ai/StartTrigger : (10-2)と同じである。

ai/SampleClock : (10-2)と同じである。



以下は全体図である。

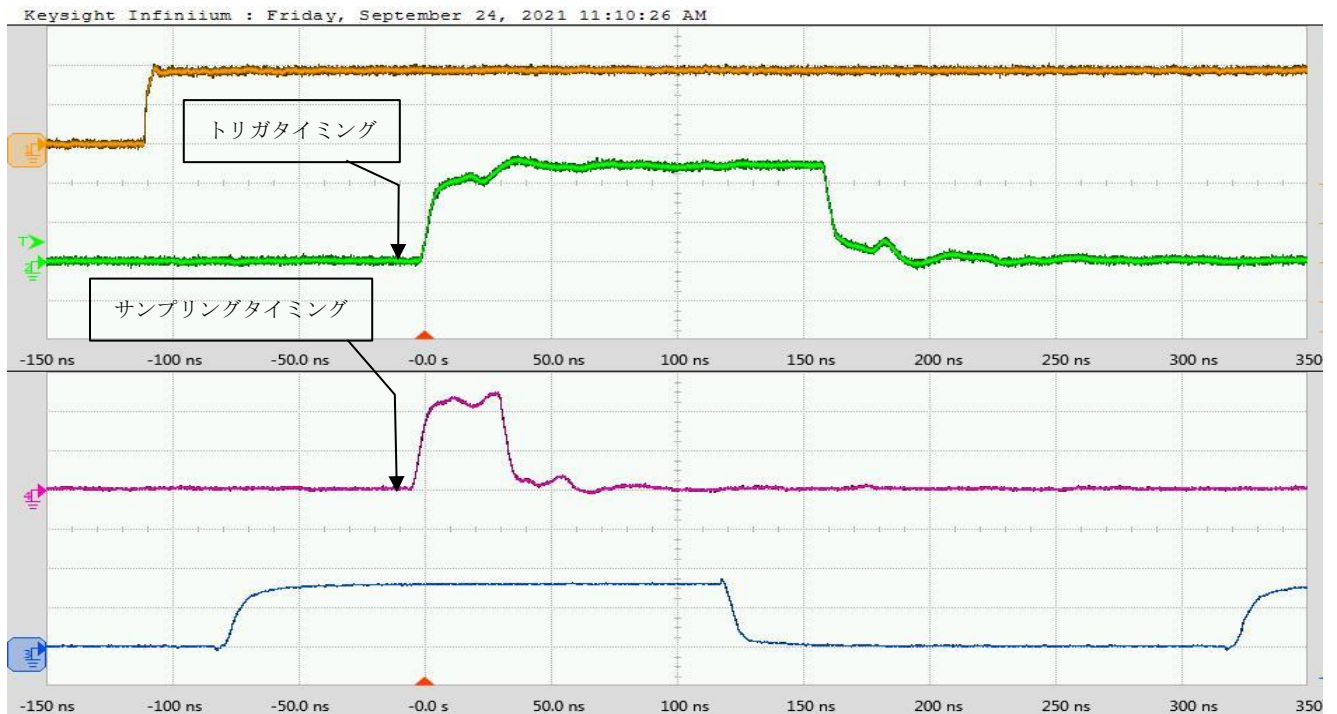


(12) External 2.5Hz, Div1, Finite PreSamples 5 : (11)のプリサンプルモード

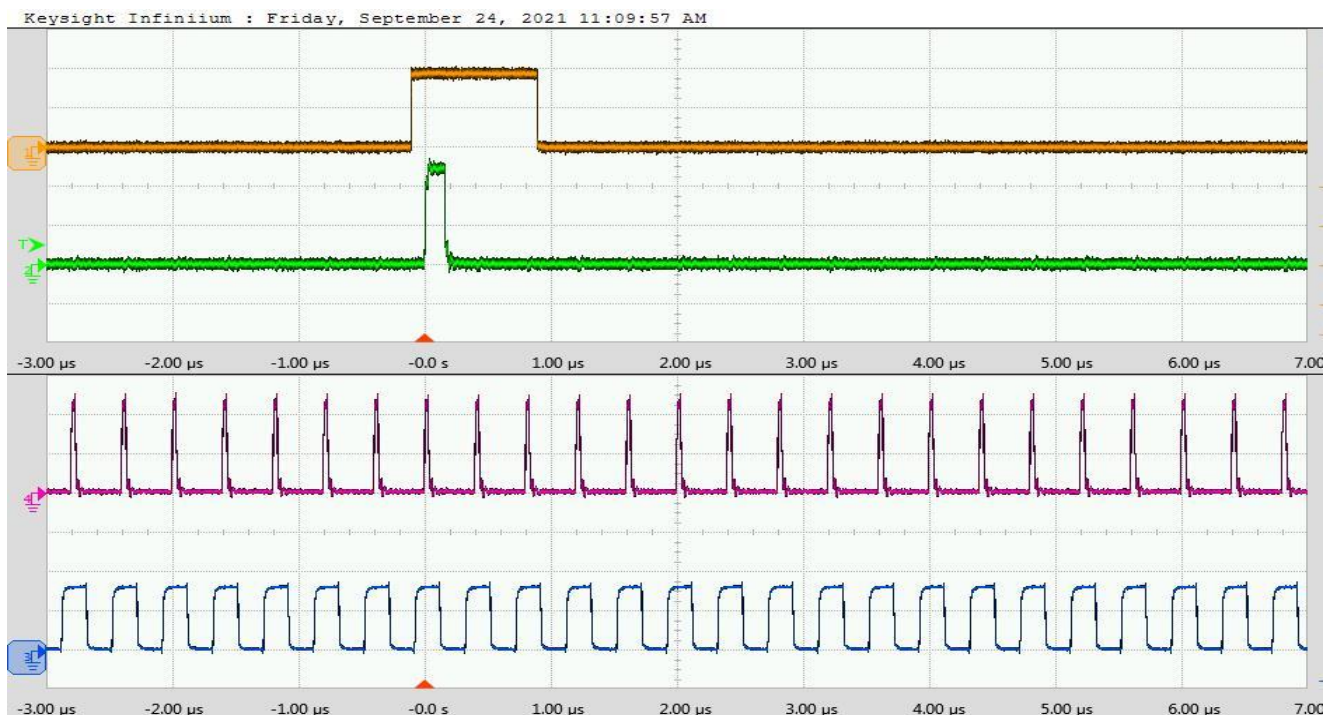
(12-1) External 2.5MHz, Div1, Finite PreSamples 5, (Sample Clock Delay 2), Sample Clock Rising

ai/StartTrigger : 外部クロックの Rising+70ns+10ns(表示上遅延)

ai/SampleClock : 外部クロックの Rising+70ns+10ns(表示上遅延)に同期



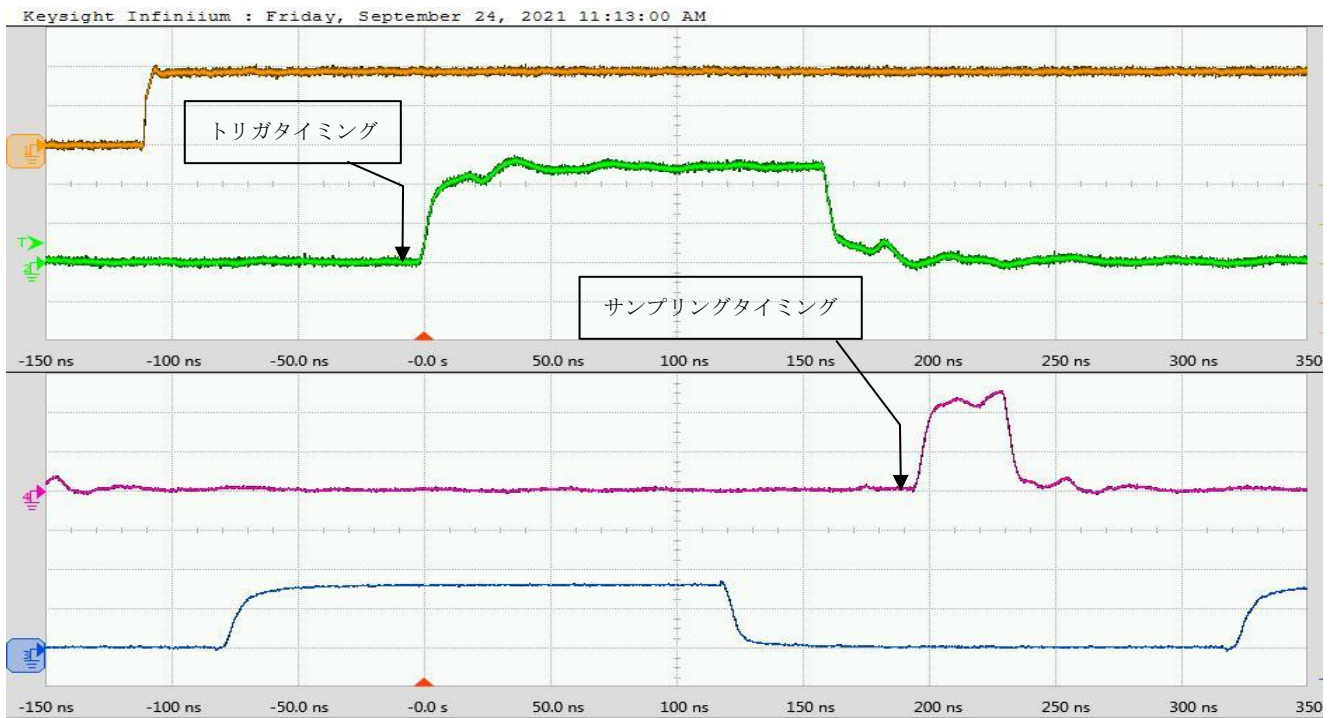
以下は全体図である。



(12-2) (12-1) Sample Clock Rising → Falling

ai/StartTrigger : 外部クロックの Rising+70ns+10ns(表示上遅延)

ai/SampleClock : 外部クロックの Falling+70ns+10ns(表示上遅延)に同期



以下は全体図である。

