**APB Timer**

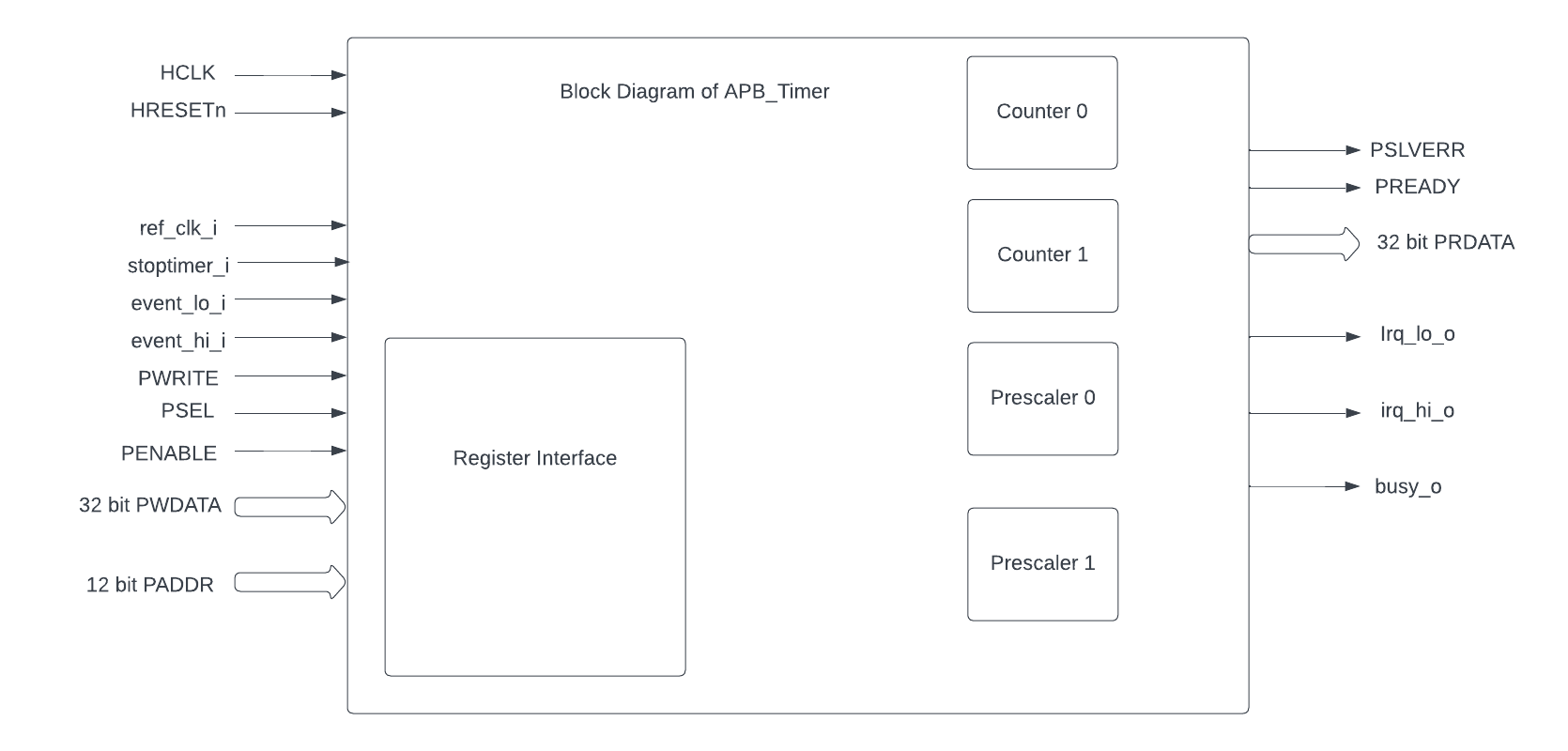
APB Timer supports two 32 bit individual timers with separate interrupt lines. APB Timer can also be configured as a 64 bit timer.

**Features**

* Multiple trigger input sources
* Configurable 32 bit or 64 bit timer
* Two 32 bit configurable prescaler
* Configurable input trigger modes
* Configurable clock gating for each timer

**Theory of Operation**

Block Diagram of APB\_Timer:



APB\_Timer can be configured in various modes like 32 bit mode or 64 bit mode.  
 32 bit mode timer:

* It supports 32 bit timer\_low and 32 bit timer\_high and they can be configured parallelly at the same time.
* timer low which has a 32 bit prescaler and 32 bit counter which will have unique input\_lo and output\_lo pins.
* timer high which has a 32 bit prescaler and 32 bit counter which will have unique input\_hi and output\_hi pins.

64 bit mode timer:

* It supports a single 64 bit timer.
* the 64 bit timer has a 32 bit prescaler and 64 bit counter. The FW has to drive both the high and low input pins
* The output will be driven in the low pin for 64 bit mode.

Assuming there is no initial count configured for the counter, basic operations of the timer are explained. The following four combinations can be run in both 32 bit mode and 64 bit mode.

Timer operation with both Prescaler and ref\_clk disabled:

* Timer module directly enables the counter to start incrementing the count for every positive edge of Hclk clock from '0' till it reaches the compare value. When the count reaches the target compare value the timer value drives the output interrupt pins if its enabled.

Timer operation with Prescaler disabled and ref\_clk enabled:

* Timer modules wait until the reference clock's edge is detected and then enable the counter to start incrementing the count for every positive edge of the reference clock from '0' till it reaches the compare value. When the count reaches the target compare value the timer value drives the output interrupt pins if its enabled.

Timer operation with Prescaler enabled and ref\_clk disabled:

* Timer module will enable the prescaler and counter in the cascaded manner that is once the prescaler target is achieved the counter will start. The prescaler will be configured and once the target compare value of the prescaler is reached then the counter will start incrementing the count for every positive edge of Hclk clock from '0' till it reaches the compare value. When the count reaches the target compare value the timer value drives the output interrupt pins if its enabled.

Timer operation with Prescaler enabled and ref\_clk enabled:

* Timer will enable the prescaler and counter in the cascaded manner that is once the prescaler target is achieved and reference clock's edge is detected the counter will start. The prescaler will be configured and once the target compare value of the prescaler is reached then the counter will start incrementing the count for every positive edge of the reference clock from '0' till it reaches the compare value. When the count reaches the target compare value the timer value drives the output interrupt pins if its enabled.

**Programming Model**

Various modes supported by the APB\_TIMER:

* One shot mode:

In this mode, the timer will be disabled completely as soon as the timer count reaches the target count for the first time.

* Compare clear mode:

In this mode, the timer will still be enabled but the timer count will be reset to '0' as soon as the timer count reaches the target compare count. So that if all the other input configurations are valid then interrupt will be driven as many times the count reaches the target compare count

Various features enabled in the APB\_TIMER:

* Mode selection of 32 bit or 64 bit counters by configuring the MODE\_64\_BIT in CFG\_REG\_LO or CFG\_REG\_HI register.
* Reset the counter value by configuring the RESET\_BIT in CFG\_REG\_LO or CFG\_REG\_HI register.
* Enable or disable the ref\_clk by configuring the REF\_CLK\_EN\_BIT in CFG\_REG\_LO or CFG\_REG\_HI register.
* Enable or disable the prescaler by configuring the PRESCALER\_BIT in CFG\_REG\_LO or CFG\_REG\_HI register.
* Enable or disable the counter to start the counting by configuring the ENABLE\_BIT in CFG\_REG\_LO or CFG\_REG\_HI register.
* Configure the Mode\_mtime bit so that in the 64 bit mode even if the IRQ\_bit is not set an interrupt is being driven when the count == compare\_value. Configure the MODE\_MTIME\_BIT in CFG\_REG\_LO or CFG\_REG\_HI register.
* Stoptimer\_i pin is used to stop the counter operation of the timer module directly.
* busy\_o pin is used to provide will be driven high if anyone of the counter is enabled.
* Overwriting the counter value directly via the by configuring the TIMER\_VAL\_LO or TIMER\_VAL\_HI register.
* Initial counter value can be configured to start the timer counter value by configuring the TIMER\_VAL\_LO or TIMER\_VAL\_HI register

**APB Timer CSRs**

### **FG\_REG\_LO offset = 0x000**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| MODE\_64\_BIT | 31:31 | RW |  | 1 = 64-bit mode, 0=32-bit mode |
| MODE\_MTIME\_BIT | 30:30 | RW |  | 1=MTIME mode Changes interrupt to be >= CMP value |
| PRESCALER\_COUNT | 15:8 | RW |  | Prescaler divisor |
| REF\_CLK\_EN\_BIT | 7:7 | RW |  | 1= use Refclk for counter, 0 = use APB bus clk for counter |
| PRESCALER\_EN\_BIT | 6:6 | RW |  | 1= Use prescaler 0= no prescaler |
| ONE\_SHOT\_BIT | 5:5 | RW |  | 1= disable timer when counter == cmp value |
| CMP\_CLR\_BIT | 4:4 | RW |  | 1=counter is reset once counter == cmp, 0=counter is not reset |
| IEM\_BIT | 3:3 | RW |  | 1 = event input is enabled |
| IRQ\_BIT | 2:2 | RW |  | 1 = IRQ is enabled when counter ==cmp value |
| RESET\_BIT | 1:1 | RW |  | 1 = reset the counter |
| ENABLE\_BIT | 0:0 | RW |  | 1 = enable the counter to count |

### 

### **CFG\_REG\_HI offset = 0x004**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| MODE\_64\_BIT | 31:31 | RW |  | 1 = 64-bit mode, 0=32-bit mode |
| MODE\_MTIME\_BIT | 30:30 | RW |  | 1=MTIME mode Changes interrupt to be >= CMP value |
| PRESCALER\_COUNT | 15:8 | RW |  | Prescaler divisor |
| REF\_CLK\_EN\_BIT | 7:7 | RW |  | 1= use Refclk for counter, 0 = use APB bus clk for counter |
| PRESCALER\_EN\_BIT | 6:6 | RW |  | 1= Use prescaler 0= no prescaler |
| ONE\_SHOT\_BIT | 5:5 | RW |  | 1= disable timer when counter == cmp value |
| CMP\_CLR\_BIT | 4:4 | RW |  | 1=counter is reset once counter == cmp, 0=counter is not reset |
| IEM\_BIT | 3:3 | RW |  | 1 = event input is enabled |
| IRQ\_BIT | 2:2 | RW |  | 1 = IRQ is enabled when counter ==cmp value |
| RESET\_BIT | 1:1 | RW |  | 1 = reset the counter |
| ENABLE\_BIT | 0:0 | RW |  | 1 = enable the counter to count |

### 

### **TIMER\_VAL\_LO offset = 0x008**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_VAL\_LO | 31:0 | RW | 0x0 | 32-bit counter value – low 32-bits in 64-bit mode |

### 

### **TIMER\_VAL\_HI offset = 0x00C**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_VAL\_HI | 31:0 | RW | 0x0 | 32-bit counter value – high 32-bits in 64-bit mode |

### 

### **TIMER\_CMP\_LO offset = 0x010**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_CMP\_LO | 31:0 | RW | 0x0 | compare value for low 32-bit counter |

### 

### **TIMER\_CMP\_HI offset = 0x014**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_CMP\_HI | 31:0 | RW | 0x0 | compare value for high 32-bit counter |

### 

### **TIMER\_START\_LO offset = 0x018**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_START\_LO | 31:0 | WS | 0x0 | Write strobe address for starting low counter |

### 

### **TIMER\_START\_HI offset = 0x01C**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_START\_HI | 31:0 | WS | 0x0 | Write strobe address for starting high counter |

### 

### **TIMER\_RESET\_LO offset = 0x020**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_RESET\_LO | 31:0 | WS | 0x0 | Write strobe address for resetting the low counter |

### 

### **TIMER\_RESET\_HI offset = 0x024**

| **Field** | **Bits** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- | --- |
| TIMER\_RESET\_HI | 31:0 | WS | 0x0 | Write strobe address for resetting the high counter |