Datetime

Table of Contents

[Overview 3](#_Toc327553386)

[Features 3](#_Toc327553387)

[Clocks 3](#_Toc327553388)

[Registers: 4](#_Toc327553389)

[Date-time Register 5](#_Toc327553390)

[Alarm Register 5](#_Toc327553391)

[Control Register 5](#_Toc327553392)

[Snapshot Register 6](#_Toc327553393)

[I/O Ports 7](#_Toc327553394)

[WISHBONE Compatibility Datasheet 9](#_Toc327553395)

# Overview

Datetime is a date and time keeping core.

# Features

* 32-bit BCD format
* Mars timekeeping option
* programmable time-of-day frequency (50/60/100Hz)
* external time-of-day clock required

# Clocks

The Datetime core uses independent bus and time-of-day (tod) clocks. The tod clock is run through a two stage synchronizer and edge detector before being used to increment the time and date.

# Registers:

The Datetime uses a 32 bit address decode, the register array is internally decoded to appear at $FFDC\_040x in the system’s memory.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| reg | Bits | R/W | Brief |  |
| 0 | HHMMSSss | RW | time in BCD format |  |
| 1 | YYYYMMDD | RW | date |  |
| 2 | HHMMSSss | RW | alarm time in BCD Format |  |
| 3 | YYYYMMDD | RW | alarm date |  |
| 4 | m ff e mmmmmmmm | W | |  |  |  | | --- | --- | --- | | Bit | Breif |  | | [7:0] | alarm care bits |  | | [8] | tod enable |  | | [10:9] | 00 = 100 Hz,  01 = 60 Hz,  10 = 50 Hz |  | | [16] | 1=Mars timekeeping |  | |  |  |  | |  |  |  | |  |
| 5 |  | W | take a snapshot |  |

## Ttime Register (FFDC0400)

This register is a record of the time in BCD format. The register may be written anytime in order to update the time. The time-of-day enable may be disabled prior to updating the register, then re-enabled afterwards, in order to avoid problems with time-of-day updates occurring while the time is being set.

In order to read the time and date, first perform a write operation to register #5 (FFDC0405) to cause a snapshot of the current date and time to be taken. Then read this register.

## Date-time Register (FFDC0401)

This register is a record of the date in BCD format. The register may be written anytime in order to update the date. The time-of-day enable may be disabled prior to updating the register, then re-enabled afterwards, in order to avoid problems with time-of-day updates occurring while the date is being set.

In order to read the time and date, first perform a write operation to register #5 (FFDC0405) to cause a snapshot of the current date and time to be taken. Then read this register.

## Alarm Register (FFDC0402)

This register contains the alarm time in the same BCD format as the time register. When the alarm date-time matches the date-time an alarm signal is set. Components of the date-time to match are set by the match bytes in the control register.

## Alarm Register (FFDC0403)

This register contains the alarm date in the same BCD format as the date register. When the alarm date-time matches the date-time an alarm signal is set. Components of the date-time to match are set by the match bytes in the control register.

## Control Register (FFDC0404)

Bits 0-7 indicate which bytes of the datetime and alam datetime registers to compare. For instance an hourly alarm may be set by clearing the year-month-day, and hours bytes.

7 – century match

6 – year match

5 – month match

4 – day match

3- hours match

2 – minutes match

1 – seconds match

0 – jiffies match

Bit 8 – ‘1’ enable time-of-day tracking, 0 disables the time-of-day updates.

Bit 10,9 – specifies the frequency of the time of day clock

00 = 100 Hz time-of-day input clock

01 = 60 Hz time of day input clock

10 = 50 Hz time of day input clock

Bit 16 – ‘1’ = keep track of Martian time and date, ‘0’ = Earth date and time.

## Snapshot Register (FFDC0405)

Writing to the snapshot register causes a snapshot of the current date and time to be taken. The snapshot time and date is then available from the Datetime register (reg #0)

Code Sample:

DATETIME EQU 0xFFDC0400

;------------------------------------------------------------------------------

;------------------------------------------------------------------------------

DisplayDatetime:

subui sp,sp,#32

sm [sp],r1/r2/r3/lr

call CursorOff

lc r2,CursorRow

lc r3,CursorCol

outw r0,DATETIME+5 ; trigger a snapshot

lw r1,#46 ; move cursor down to last display line

sc r1,CursorRow

lw r1,#64

sc r1,CursorCol

inw r1,DATETIME ; get the snapshotted date and time

call DisplayWord ; display on screen

sc r2,CursorRow ; restore cursor position

sc r3,CursorCol

call CalcScreenLoc

call CursorOn

lm [sp],r1/r2/r3/lr

ret #32

# I/O Ports

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Wid | I/O | Description |  |
| rst\_i | 1 | I | This is the active high reset signal |  |
| clk\_i | 1 | I | system bus clock |  |
| cyc\_i | 1 | I | cycle active |  |
| stb\_i | 1 | I | data strobe |  |
| ack\_o | 1 | O | data transfer acknowledge |  |
| we\_i | 1 | I | write cycle |  |
| sel\_i | 8 | I | byte lane selects |  |
| adr\_i | 34 | I | decode / register address |  |
| dat\_i | 32 | I | data input |  |
| dat\_o | 32 | O | data output |  |
| tod | 1 | I | tod pulse input (eg 100 Hz) |  |
| alarm | 1 | O | alarm match output |  |
|  |  |  |  |  |
|  |  |  |  |  |

# WISHBONE Compatibility Datasheet

The Datetime core may be directly interfaced to a WISHBONE compatible bus.

|  |  |  |
| --- | --- | --- |
| WISHBONE Datasheet  WISHBONE SoC Architecture Specification, Revision B.3 | | |
|  |  | |
| Description: | Specifications: | |
| General Description: | Datetime – date and time keeping | |
| Supported Cycles: | SLAVE, READ / WRITE  SLAVE, BLOCK READ / WRITE  SLAVE, RMW | |
| Data port, size:  Data port, granularity:  Data port, maximum operand size:  Data transfer ordering:  Data transfer sequencing | 32 bit  32 bit  32 bit  Little Endian  any (undefined)  must write register #3 before reading #0 | |
| Clock frequency constraints: | 50/60/100 Hz time of day clock | |
| Supported signal list and cross reference to equivalent WISHBONE signals | Signal Name:  ack\_o  adr\_i(63:0)  clk\_i  dat\_i(63:0)  dat\_o(63:0)  cyc\_i  stb\_i  we\_i | WISHBONE Equiv.  ACK\_O  ADR\_I()  CLK\_I  DAT\_I()  DAT\_O()  CYC\_I  STB\_I  WE\_I |
| Special Requirements: |  | |