

# CAST

## ATAIF-SD

### Parallel Advanced Technology Attachment (PATA) Software Driver

The ATAIF-SD complements that ATAIF controller core to provide a complete solution for controlling mass storage media using the Parallel ATA interface.

The driver provides a software abstraction of the controller functions enabling smooth integration in a target application. It allows the user to easily access high capacity storage media, such as Hard Disk Drives and Compact Flash cards, without detailed knowledge of the controller interface.

Transfer operations can be executed using the Programmed Input/Output (PIO), Multiword DMA (MDMA) or Ultra-DMA ATA transfer modes. All operations can be performed using transaction-level functions (consistent with eCos OS driver model) or by simple-to-use API. The driver supports both the ATA device and the DMA controller interrupts generated by the ATAIF core.

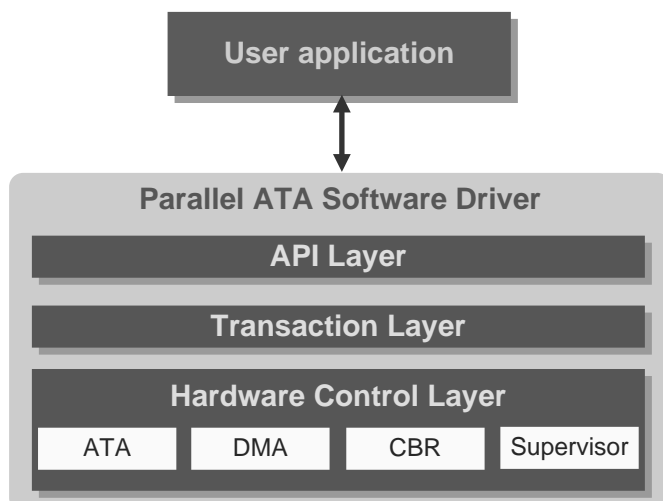
Supporting two channels (master and slave) makes the operations on two drives significantly easier. Allowing for dynamic timings configuration, the driver allows each device on the cable to transfer data at its own best speed.

The software driver is provided in C source code, and it therefore portable to any processor. It has been tested and verified with the ATAIF core on NIOS-II and Freescale's 68K and Coldfire processors.

#### Applications

- Personal Digital Assistants
- Digital cameras
- Network disk servers
- USB mass storage devices
- Picture tanks
- Multimedia players
- Set-top boxes
- Stand-alone DVD recorders
- Compact Flash card readers
- Internet terminals

#### Block Diagram



#### Features

- **Easy to use API interface**
  - API consisting of just four functions: APIIni, APIRead, APIWrite, APICatchError
  - Control parameter passing by structures
  - Transaction layer interface based on eCos driver model
  - Easy adaptation to use as an eCos driver
  - Easy to use and flexible
- **Thorough errors control**
  - Uniform numeration of all errors
  - Error descriptions
  - Defined macros to provide error handling
- **Flexible Debug capability**
  - Independent debug module
  - Provides functions to support debugging by UART interface (or other output interface)
- **All ATA-IF transfer modes**
  - Register PIO mode
  - All PIO modes
  - Multiword-DMA modes
  - Ultra-DMA ATA modes
- **Read/write sector operations**
  - Variable buffer length
  - LBA address mode
- **Flexible device timings**
  - Dynamically timings configurations
  - Independent configuration of timings for all supported modes
- **Identify device block read support**
- **Interrupts support**
  - Support for ATA device interrupt handling
  - Support for DMA controller interrupt handling

## Functional Description

The ATAIF Software Driver is partitioned into modules as shown in the block diagram and described below.

### API Layer

The API layer provides high level operations and abstract data types. It receives the abstract data types and API functions parameters, carries out appropriate transaction, gets results and converts them to format acceptable by the user. The following actions are performed by the API layer:

- Read and write data operations
- Read device identify block
- ATAIF initialization

### Transaction Layer

The Transaction Layer supervises the disk and ATAIF processes by synchronizing the consecutive operations carried through API (user, process, etc.). The built-in functionality facilitates the use of software package in a multi-tasking environment such as RTOS. The main tasks provided by the transaction layer are:

- DMA data transfer from/to disc
- PIO data transfer from/to disc
- Getting Identification Device Block
- Disc command (e.g. set features, set DMA mode, etc.)
- ATAIF component and subcomponents (discs devices) initialization.

### Hardware Control Layer

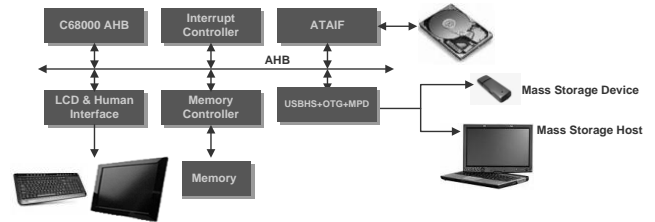
The Hardware Control Layer functionality is based on four blocks that interact directly with the ATAIF core registers.

The blocks are:

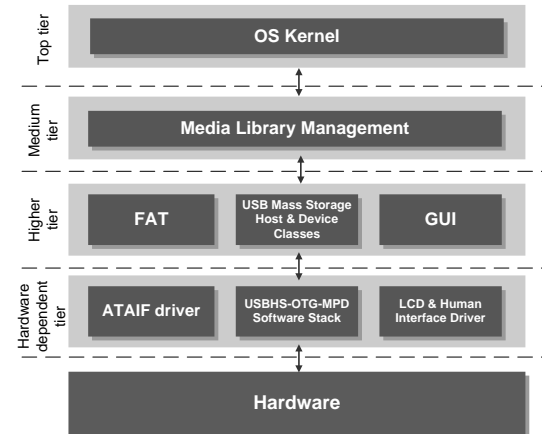
- ATA – controls transfer between the storage device and the ATAIF controller core. It can start or stop data transmission between storage device and the internal FIFO of the ATAIF controller core.
- DMA – supervises the transfers between the ATAIF controller core and the system memory
- CBR – controls the read/write CBR registers; it reads and writes the Control Block Registers and the Command Block Registers in the storage device. The CBR verifies IDE device operations and allows to check its status; the blocks of registers are described in the ATAIF-7 standard
- Supervisor – enables/disables ATAIF interrupts, re-sets ATAIF component, and gets information on ATAIF status

## Example Application

The figure below illustrates an example application that provides access to a Hard Disk Drive via USB. The example application uses an ATAIF controller connected to the Hard Disk Drive, an Interrupt Controller, the C68000-AHB Micro-processor, a Memory Controller, a USB OTG controller, an LCD and a keyboard as a Human Interface.



The software part of this example application contains the hardware-dependent tier, a medium tier (i.e. FAT, GUI, USB Mass Storage Host & Device Classes), and a higher tier (i.e. Media Library Management and Top tier: OS Kernel). The hardware-dependent tier includes ATAIF Software Driver, the Software Stack for CAST's USBHS-OTG-MPD core, and the LCD & Human Interface Driver,



## Verification

This software was created and tested using both Nios™ II Integrated Development Environment (IDE) and the TASKING 68K/ColdFire Software Development Tools.

## Deliverables

- C source code of all ATAIF modules
- Design Specification
- Reference project created in both Nios™ II IDE and the Tasking environment

## Related Products

**ATAIF - ATA/IDE/ATAPI ATA-8 Interface Core**