# Sirius TagBoard

Revision: 0.61

July 24, 2014

| version | date | author | description | sirLinux ver |
|---|---|---|---|---|
| 0.60 | 25/05/2014 | Marco Trentarossi | first delivery | - |
| 0.61 | 23/07/2014 | Francesco Trentarossi | FCC / Industry Canada statements | - |

Table 1: revision history

# Contents

# Chapter 1

# General information

## 1.1 Operational description

Sirius DDT TAGBOARD is an industrial computer, with real time capabilities, based on MPC5125; directly powered at 24Vdc.

The operating system that comes with it is UNIX-LIKE and is based on the linux kernel. The distribution is an OpenEmbedded / Yocto derivative and allows the management of applications through packets.

Thanks to the Xenomai scheduler, you can create tasks with real-time characteristics entirely in userspace.

Peripherals include two Ethernet devices, three CAN channels and two optocoupled serial ports.

# Chapter 2

# Hardware

## 2.1 Introduction

Sirius TagBoard DDT is a real time industrial computer.

The microprocessor is a MPC5125 PowerPC with a clock frequency of 400MHz, coupled with a 16-bit PIC co-processor for the management of some non-critical tasks.

There are 256MB of DDR2 RAM, 8MB NOR for the kernel and boot, 2GB eMMC for the file system, 128kB SRAM with battery backup and 1GB or 4GB NAND.

DDT has following communication interfaces:

- 3 optocoupled CANopen ports
- 1 10/100 ethernet port
- 1 three way 10/100 ethernet switch
- 2 optocoupled serial ports, configurables as RS232, 485 o 422 according to wiring
- 2 USB 2.0 ports
- 1 programmable dot matrix informative panel
- 1 microSD card slot

## 2.2 Product images

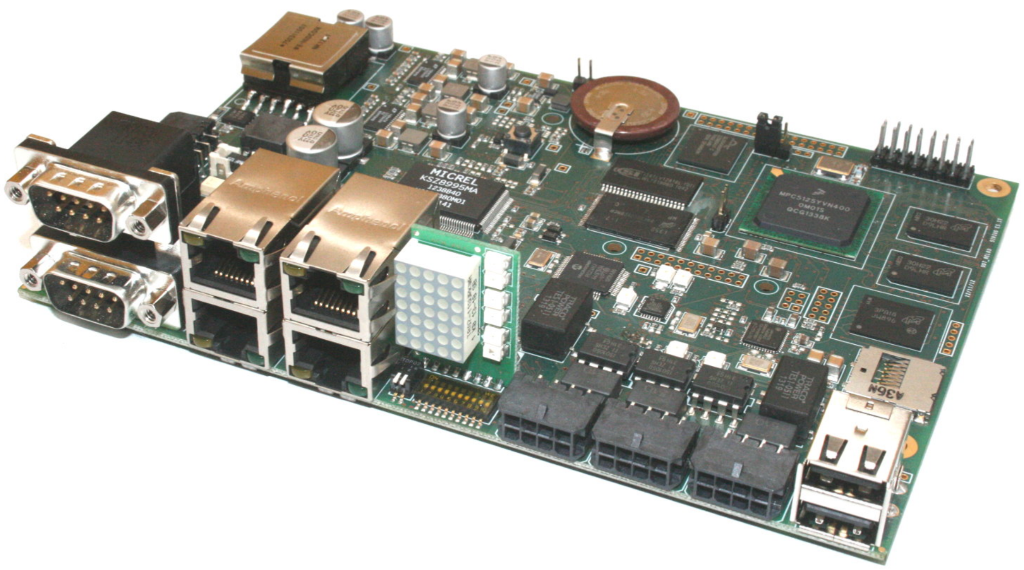


Figure 2.1: DDT-A

## 2.3 Configuration

| | *Description* |
|---:|:---:|
| **CPU** | CPU MPC5125 400MHz |
| **COPROCESSOR** | PIC 24HJ64GP506 16bit |
| **RAM** | 256 MB DDR2 |
| **iNTERNAL STORAGE** | 1-4GB NAND Flash<br>2GB eMMC<br>128kB SRAM |
| **STORAGE** | microSD card slot |
| **COMMUNICATION** | 4 RJ45 Ethernet ports<br>2 USB 2.0 type A ports<br>2 RS232/485/422 COM ports<br>3 CANopen ports |
| **POWER SUPPLY** | 24Vdc optocoupled |
| **ADDITIONAL FEAT.** | 7x5 Dot matrix display |

## 2.4 Specifications

### 2.4.1 Power supply

| | *DDT-A* |
|---|---|
| **Input Voltage** | 16-34Vdc |

### 2.4.2 UART Ports

| | *DDT-A* |
|---|---|
| **Number** | 2 |
| **Type** | RS232 - RS485 - RS422 cable select |

### 2.4.3 Ethernet Ports

| | *DDT-A* |
|---|---|
| **ETH0** | 1x RJ45 Ethernet port 10/100 |
| **ETH1** | 3x RJ45 Ethernet switch 10/100 |

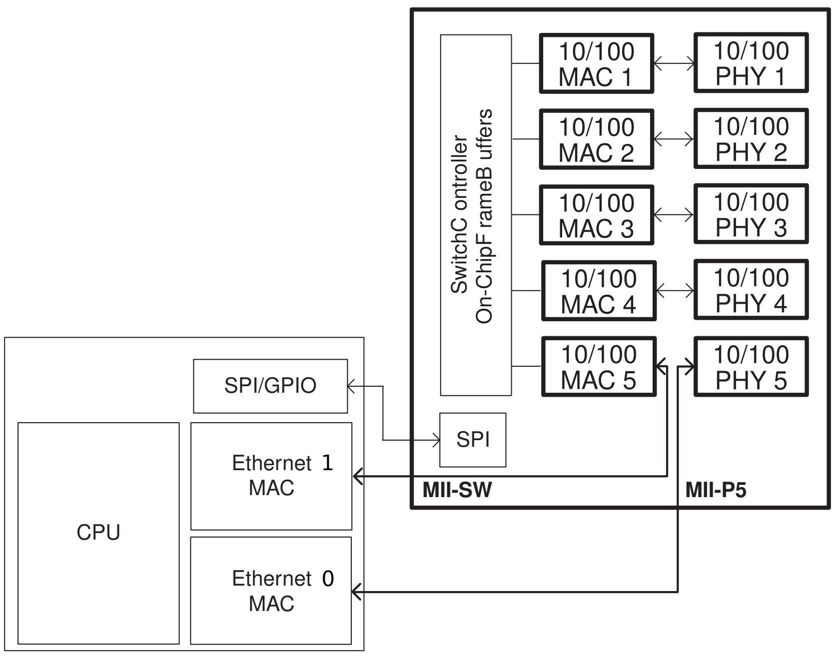


Figure 2.2: Ethernet configuration

### 2.4.4 CAN Ports

| | *DDT-A* |
|---|---|
| **Number** | 3 |
| **Type** | 3 CANopen Master/Slave selectable |
| **Signals** | CANH, CANL, GND |
| **Insulation** | CAN interface circuits and +5 Vdc supply for CAN are optically insulated from CPU |
| **Format** | CAN V2.0b physical layer for high-speed connection compliant |
| **Data** | in according CANopen CIA DS301 |
| **Address selection** | 1 channel is determined by dip-switch |
| **Stub** | 121 ohm hardware selectable |

### 2.4.5 Status Indicators

| | *DDT-A* |
|---|---|
| **CAN status** | green and red leds, in according with CAN indicator specification DR303-3 |
| **NAND status** | Solid red when NAND is used |
| **Power ON** | Solid green when power on |
| **PIC status** | Flashing red when Co-processor is operating |
| **ACM matrix** | Customizable 5x7 led dot matrix |

### 2.4.6 Protections

| | *DDT-A* |
|---|---|
| **Reverse polarity** | logic circuit, serie diode<br>power circuit, serie diode |
| **Short circuit** | logic circuit, 500mA fuse |

### 2.4.7 Mechanical & Environmental

| | *DDT-A* |
|---|---|
| **Size (L x W)** | 160 x 114 mm |
| **Height** | 15 mm |
| **Weight** | 161 g |
| **Ambient temperature** | 0 to +45 ºC operating, -40 to +85 ºC storage |
| **Humidity** | 0 to 95%, non condensing |

### 2.4.8 Agency Conformance

| | *DDT-A* |
|---|---|
| **CE** | CE compliant |
| **61000_6_4** | Generic standards - Emission Standard for industrial environments |
| **61000_6_2** | Generic standards - Immunity for industrial environments |
| **ROHS** | ROHS Compliant |

## 2.5 Lights

There are three active leds to indicate the status of the board.

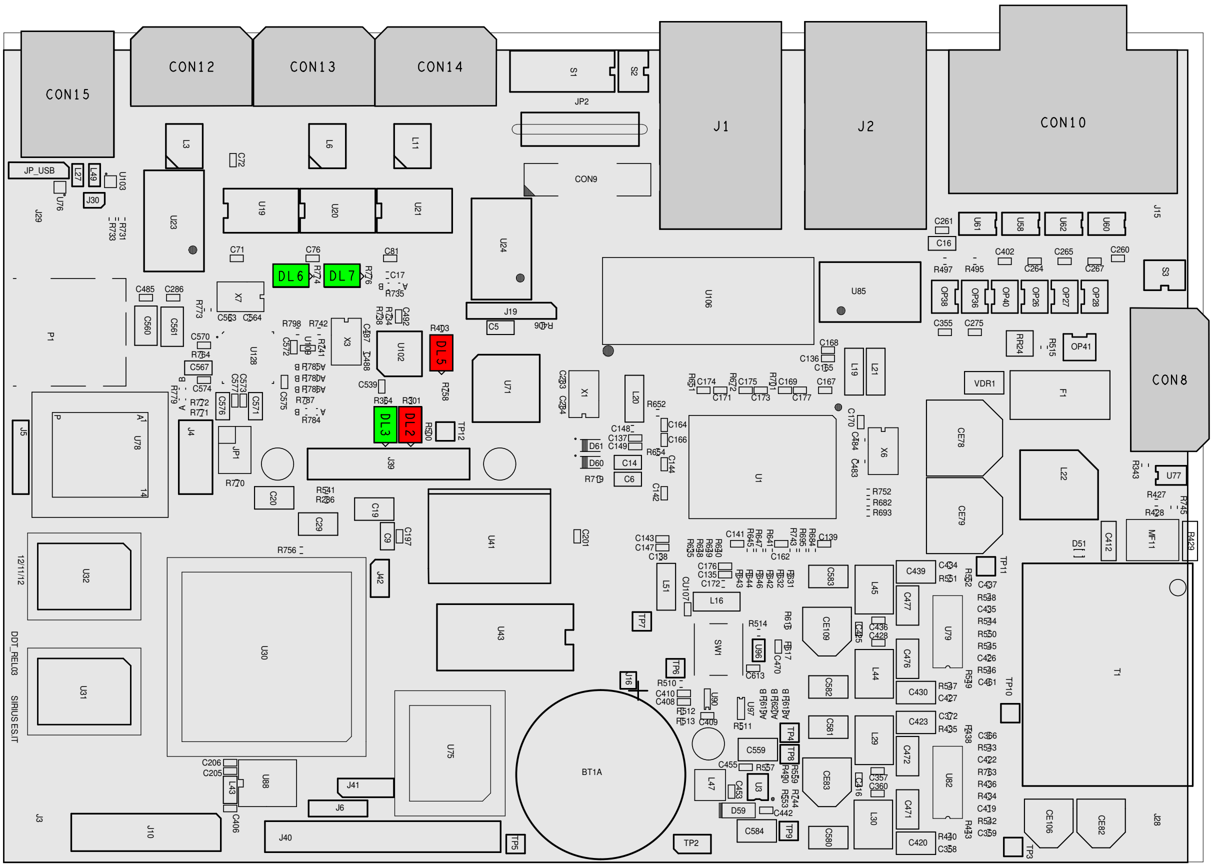


Figure 2.3: Lights position

| LED | Color | Description |
|---|---|---|
| DL2 | RED | On when accessing NAND memory |
| DL3 | GREEN | On when 5Vdc power supply is present |
| DL5 | RED | Blinks when co-processor is operating |
| DL6 | GREEN | - |
| DL7 | GREEN | - |

## 2.6 ACM informative panel

There is a programmable information panel called ACM. Its location is shown in Figure 2.4

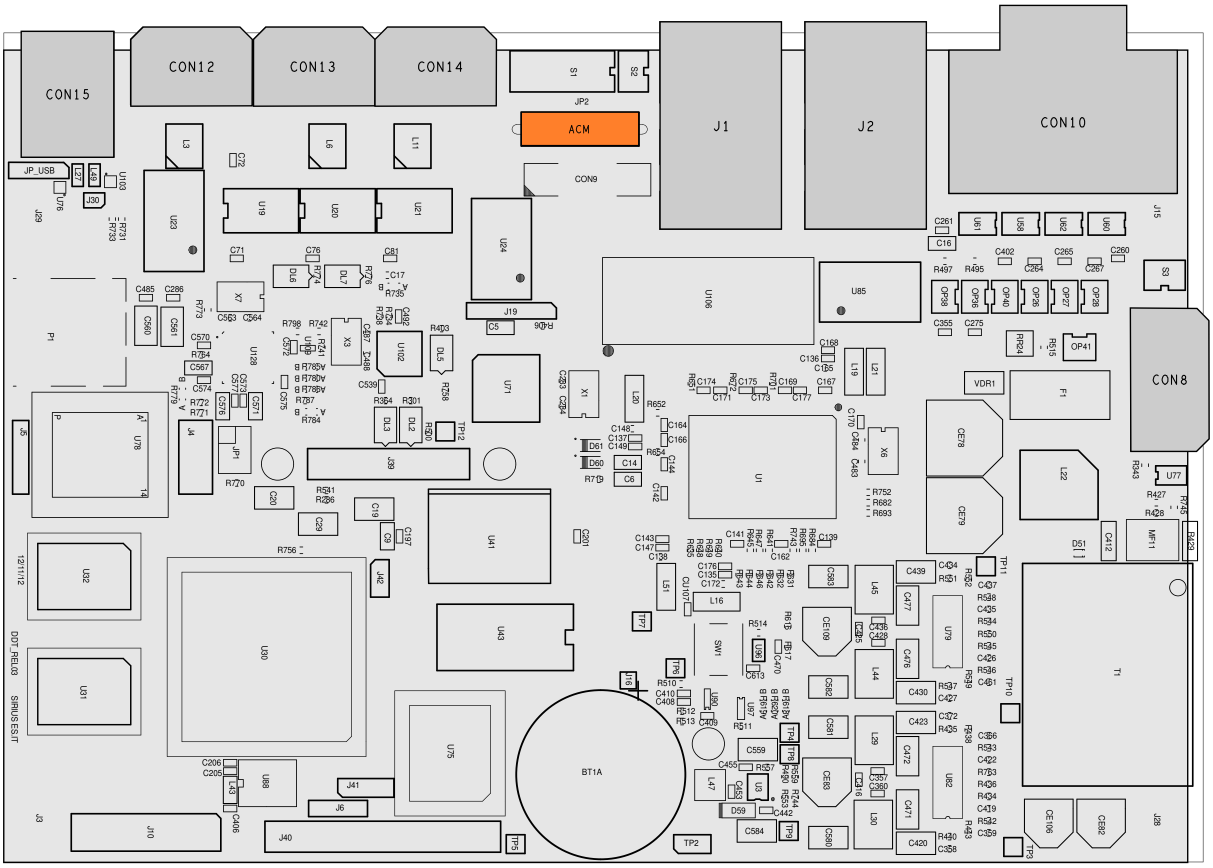


Figure 2.4: ACM panel position

Information panel is made of:

- 5x7 dot matrix display with customizable messages
- Three led pairs that provide CAN status information

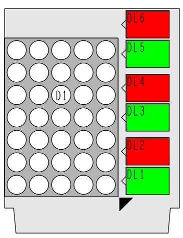


Figure 2.5: ACM panel view

| LED | Colore | Description |
|---|---|---|
| DL1 | GREEN | CAN 1 Status Led |
| DL2 | RED | CAN 1 Error Led |
| DL3 | GREEN | CAN 2 Status Led |
| DL4 | RED | CAN 2 Error Led |
| DL5 | GREEN | CAN 3 Status Led |
| DL6 | RED | CAN 3 Error Led |

The specifications of lights during operation of the device reflect what indicated in the document CiA DR 303-3 v1.2.

5x7 dot matrix panel D1 is programmable in order to create custom messages.

Co processor operates the matrix following a map of bits positioned in the CPU.

## 2.7 Selectors

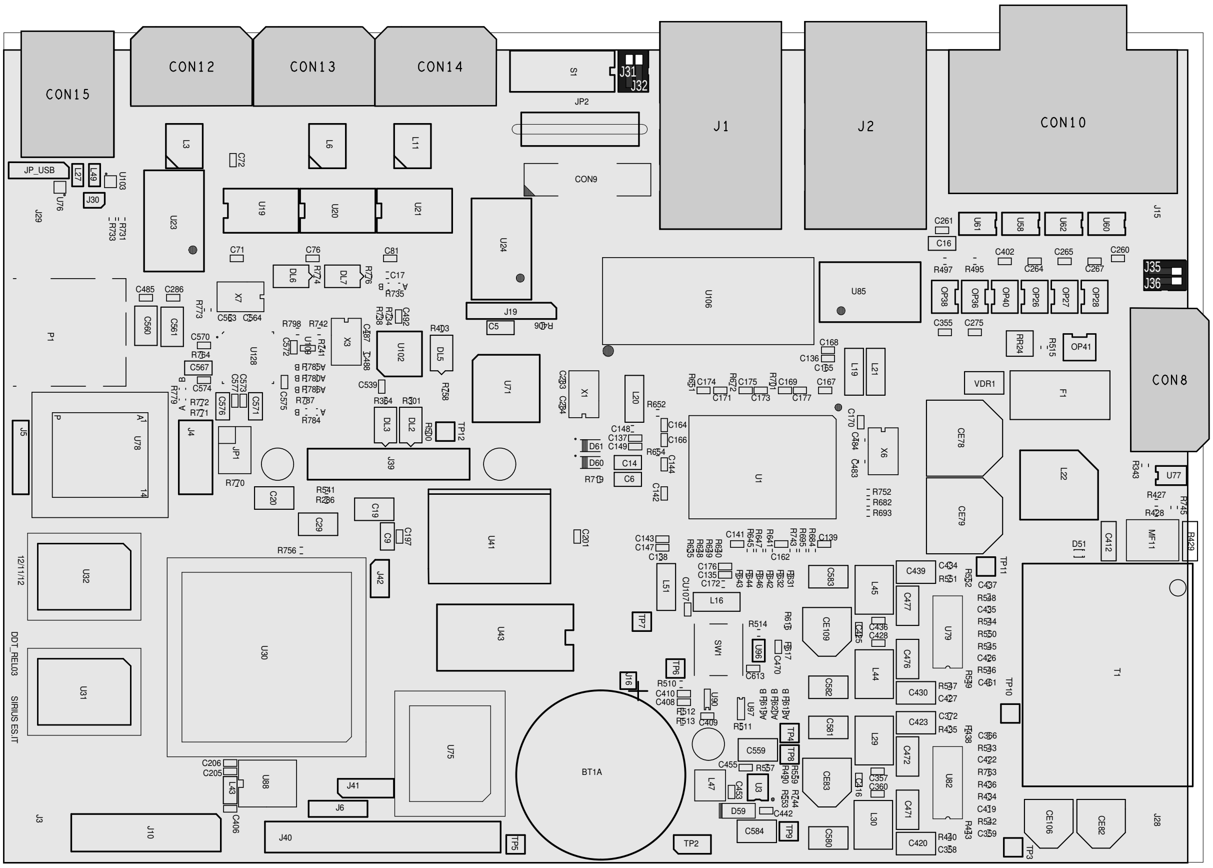


Figure 2.6: Location of selectors

| Selector | *Description* |
|---|---|
| J31 | CAN 1 terminating resistor |
| J32 | CAN 2 terminating resistor |
| J35 | UART0 terminating resistor when in RS485 mode |
| J36 | UART1 terminating resistor when in RS485 mode |

## 2.8 Micro SD slot

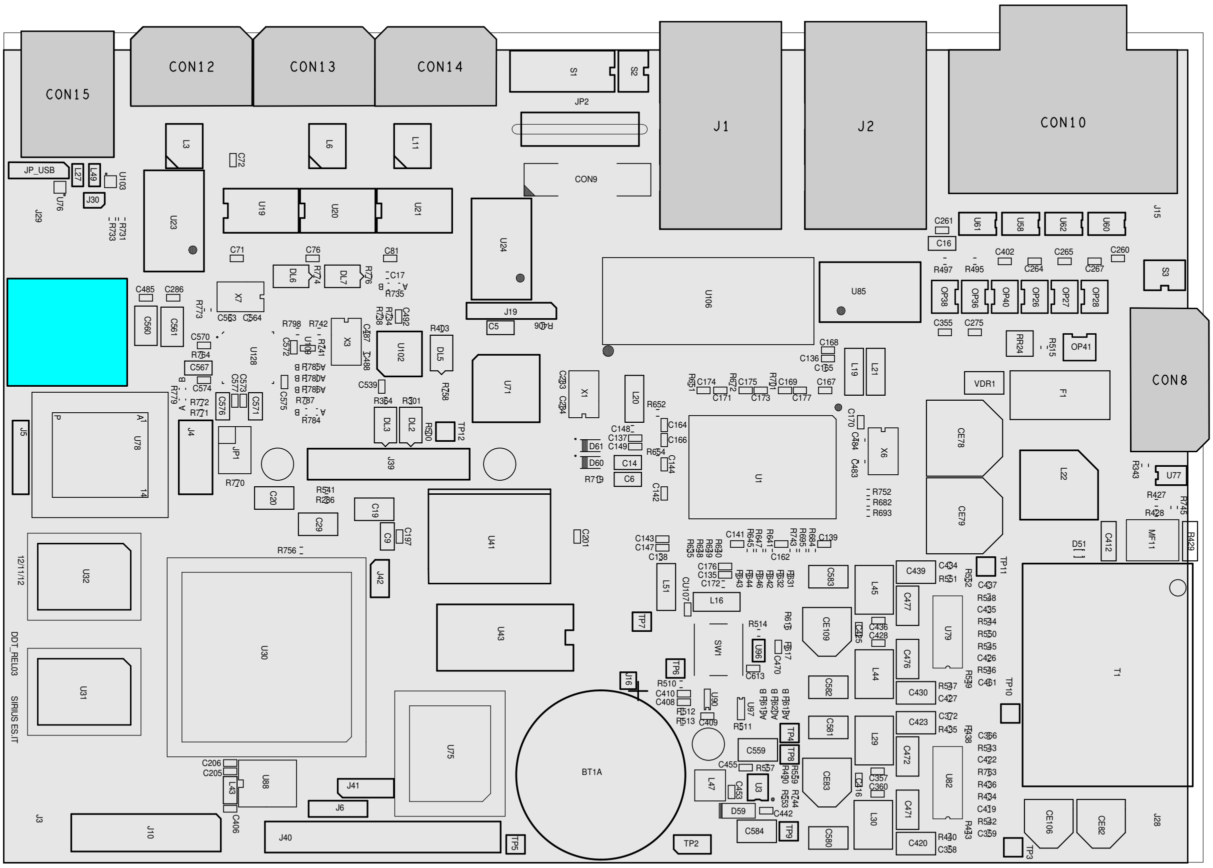

Figure 2.7: MicroSD slot location

The position of the MicroSD cage slot is shown in figure 2.7

## 2.9 Connections

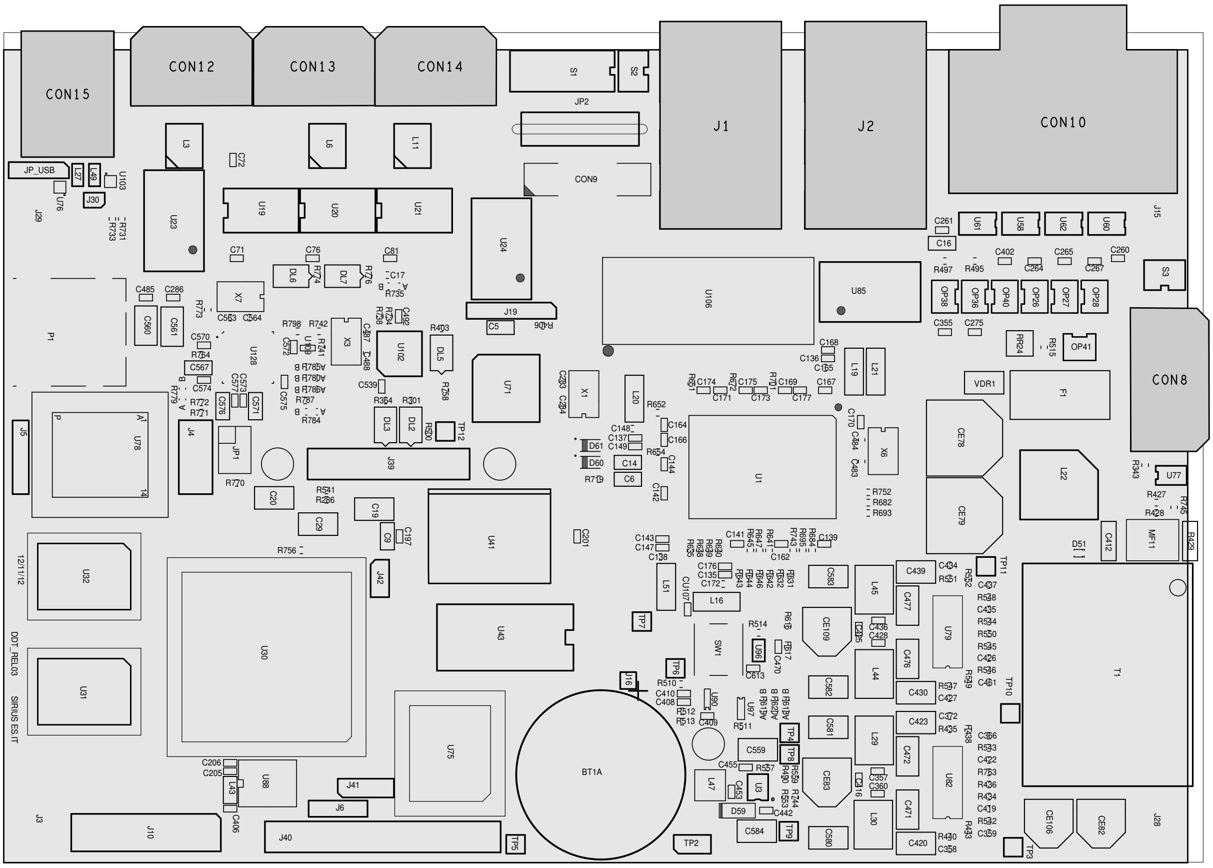


Figure 2.8: DDT-A connectors

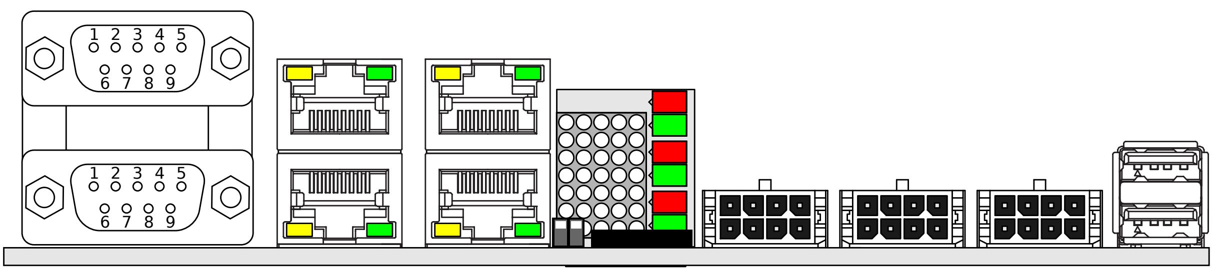


Figure 2.9: DDT-A communication ports

The connectors are distributed on the perimeter of the card occupying two sides. All coomunication ports are located on the long side of the card, while the power supply connector is located laterally.

### 2.9.1 CON8 - Power supply

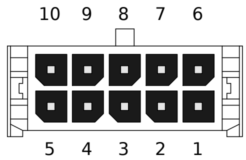


Figure 2.10: MICROFIT 3.0 430451000

| *pin* | *name* | *type* | *description* |
|---|---|---|---|
| 1 | comune | - | 0V power supply reference |
| 2 | 24V | IN | 24V power supply |
| 3 | GND | - | ground |
| 4 | NC | - | not connected |
| 5 | WDO - | OUT | negative side watchdog |
| 6 | comune | - | 0V power supply reference |
| 7 | 24V | IN | 24V power supply |
| 8 | GND | - | ground |
| 9 | NC | - | not connected |
| 10 | WDO + | OUT | positive side watchdog |

### 2.9.2 CON10 - Serial ports

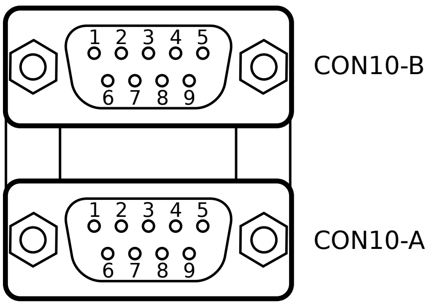


Figure 2.11: DSUB 9 PINS

| pin | name | type | description |
|---|---|---|---|
| 1 | NC | - | Not connected |
| 2 | RXD | IN | Receive data 232 |
| 3 | TXD | OUT | Transmit data 232 |
| 4 | NC | - | Not connected |
| 5 | GND | - | Ground |
| 6 | VC RX + | IN | Receive data 422 + |
| 7 | VC TX + | OUT | Transmit data 422 + |
| 8 | VC RX - | IN | Receive data 422 - |
| 9 | VC TX - | OUT | Transmit data 422 - |

RS485 communication is cable dependent

### 2.9.3 CON12- CANopen CAN 0

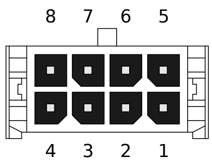


Figure 2.12: MICROFIT 3.0 430450800

| pin | name | type | description |
|---|---|---|---|
| 1-5 | CH+ | IN/OUT | CAN high |
| 2-6 | CH- | IN/OUT | CAN low |
| 3-7 | CAN reference | - | 0V CAN reference |
| 4-8 | CAN reference | - | 0V CAN reference |

### 2.9.4 CON13 - CANopen CAN 1

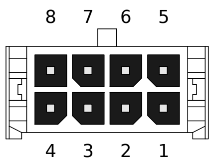


Figure 2.13: MICROFIT 3.0 430450800

| *pin* | *name* | *type* | *description* |
|---|---|---|---|
| 1-5 | CH+ | IN/OUT | CAN high |
| 2-6 | CH- | IN/OUT | CAN low |
| 3-7 | CAN reference | - | 0V CAN reference |
| 4-8 | CAN reference | - | 0V CAN reference |

### 2.9.5 CON14 - CANopen CAN 2

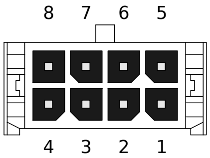


Figure 2.14: MICROFIT 3.0 430450800

| *pin* | *name* | *type* | *description* |
|---|---|---|---|
| 1-5 | CH+ | IN/OUT | CAN high |
| 2-6 | CH- | IN/OUT | CAN low |
| 3-7 | CAN reference | - | 0V CAN reference |
| 4-8 | CAN reference | - | 0V CAN reference |

### 2.9.6 CON15 - Double USB 2.0 connector

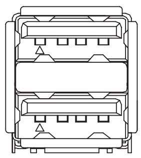

Figure 2.15: USB 2.0 TYPE A

| *pin* | *name* | *type* | *description* |
|---|---|---|---|
| 1 | Vbus | OUT | + 5Vdc |
| 2 | D - | IN/OUT | Data - |
| 3 | D + | IN/OUT | Data + |
| 4 | GND | - | Ground |

### 2.9.7 J1/J2 - Ethernet Hub

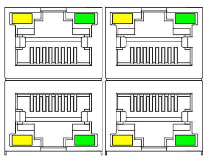

Figure 2.16: 10/100 RJ45 ETHERNET FOUR PORTS

| pin | name | type | description |
|---|---|---|---|
| 1 | TX + | OUT | Transmit data + |
| 2 | TX - | OUT | Transmit data - |
| 3 | RX + | IN | Receive data + |
| 4 | NC | - | Not connected |
| 5 | NC | - | Not connected |
| 6 | RX - | IN | Receive data + |
| 7 | NC | - | Not connected |
| 8 | NC | - | Not connected |

## 2.10 Dip-switch

Dip-switch S location is shown in figure.

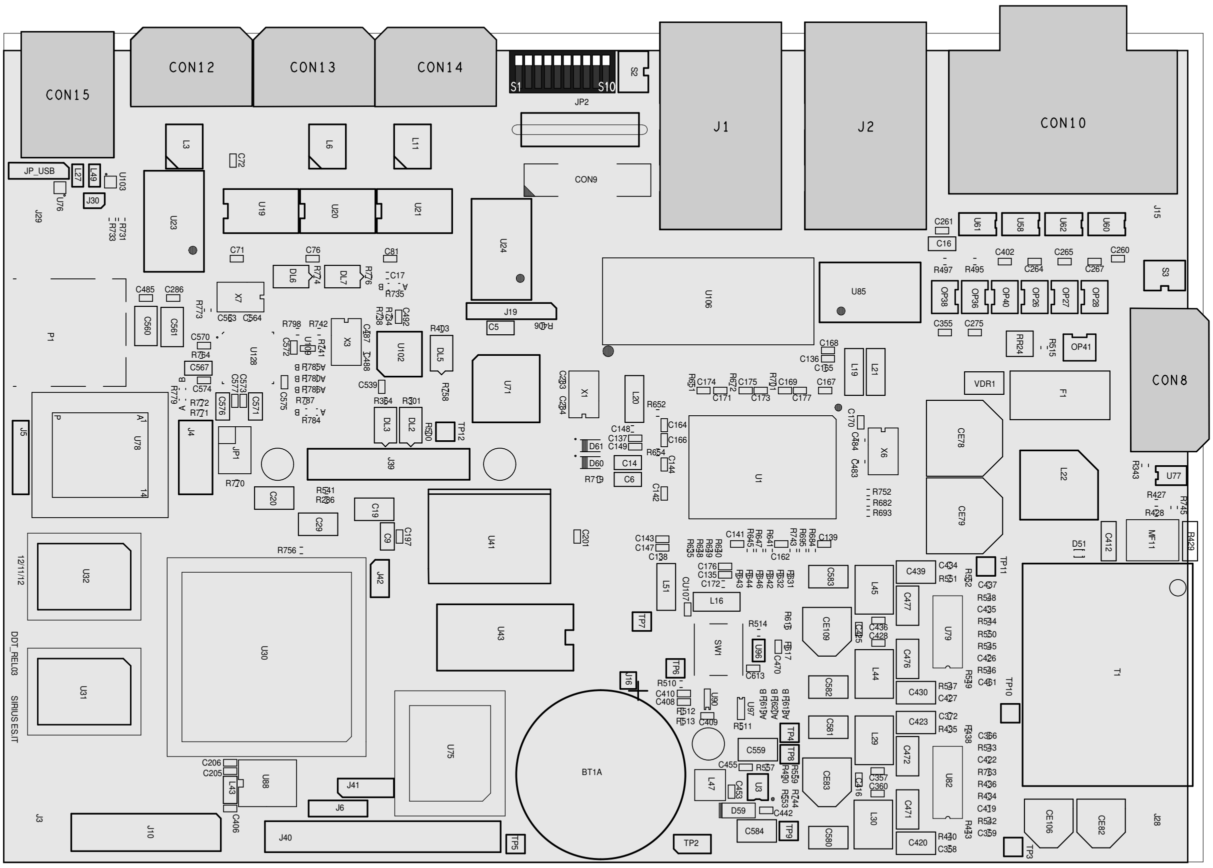


Figure 2.17: Posizione del dip-switch su DDT-A

### 2.10.1 Address selection

Using S1 selector it is possible to change the CAN address of the device.

Assigned address is stored in variable CobId.

Address can be a number from 0 to 127. Every CAN node must have an unique address.

Follows bit weight:

| ***Switch*** | ***Peso*** |
|---|---|
| S2 ON | 1 |
| S3 ON | 2 |
| S4 ON | 4 |
| S5 ON | 8 |
| S6 ON | 16 |
| S7 ON | 32 |
| S8 ON | 64 |

Table 2.1: dip-switch weight

### 2.10.2 Communication speed selection

Using S1 selector it is possible to change the communication speed of the device.

Assigned speed is stored in variable CobBR.

Speed can be 125, 250, 500 or 1000 kbps.

| ***S10*** | ***S9*** | ***Baud rate kbps*** |
|---|---|---|
| OFF | OFF | 125 |
| OFF | ON | 250 |
| ON | OFF | 500 |
| ON | ON | 1000 |

Table 2.2: baud rate selection

## FCC Statement

*Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.*

*Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.*

---

## Industry Canada statement

*This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions:*

1. *this device may not cause interference, and*
2. *this device must accept any interference, including interference that may cause undesired operation of the device.*

*Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes :*

1. *l'appareil ne doit pas produire de brouillage, et*
2. *l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.*

*CAN ICES-3 (A)/NMB-3(A)*

# Chapter 3

# Software

## 3.1 Operating system

With the Sirius TagBoard DDT comes a "Unix-like" operating system based on the Linux kernel, version 3.9.4. The current release of the kernel ensures complete compatibility with all parts of the TagBoard.

The kernel image is compliant with the GNU GPLv2 license, and is then distributed along with its source code.

The Linux kernel is designed to be compatible with the IEEE POSIX interprocess communication. This allows users to compile most of the existing Unix programs and run them.

The distribution is based on the Yocto Project and allows the management of applications through packets, as in Linux distributions for desktop systems.

The applications developed in userspace can use all the system libraries provided by OpenEmbedded / Yocto. These libraries have been released under the LGPL license, so that the developer has the option to assign the preferred license to his own software. In userspace, there are no implications or dependencies enforced with proprietary software.

The operating system integrates Xenomai scheduler to guarantee real-time performance.

## 3.2 Device Drivers

The operating system integrates the drivers for all devices on the Tag Board.

The device drivers are provided as modules and are also compatible with the GPLv2 license. In the event that it is necessary to design and implement a new driver, will be evaluated from time to time with the type of license which can be distributed, in accordance with the client.

## 3.3 Real-Time constraints

To ensure to processes desired real-time constraints, in addition to the Linux traditional preemptive scheduler, is also present a further scheduler called Xenomai. This solution provides a set of APIs that allow you to create processes with real-time characteristics entirely in userspace, greatly simplifying the implementation and debugging of the entire project. To manage priorities between tasks in real time, Xenomai scheduler supports both FIFO and Round-Robin policies.

Xenomai makes the Linux kernel a "hard real time" operating system. This allows the designer to develop real-time tasks as abstract cells, independent from the non real time system part, so that those have no coexistence issues (race conditions, starvation, ...).

In the practice Xenomai provides determinism on the scheduling of real-time tasks and interrupt response with acceptable latency for real time applications, in the order of 100us.

Xenomai also solves the problem of the microprocessor overload caused by the real-time scheduler, which may impair the ability to perform communication with, for example, other peripheral devices, serial ports or on the CAN bus.

It is therefore possible to develop real-time device drivers (RTDM API) providing deterministic access to the communication pipes, maximizing the speed.

The architecture and implementation of the Xenomai APIs follow the logic diagram shown in figure

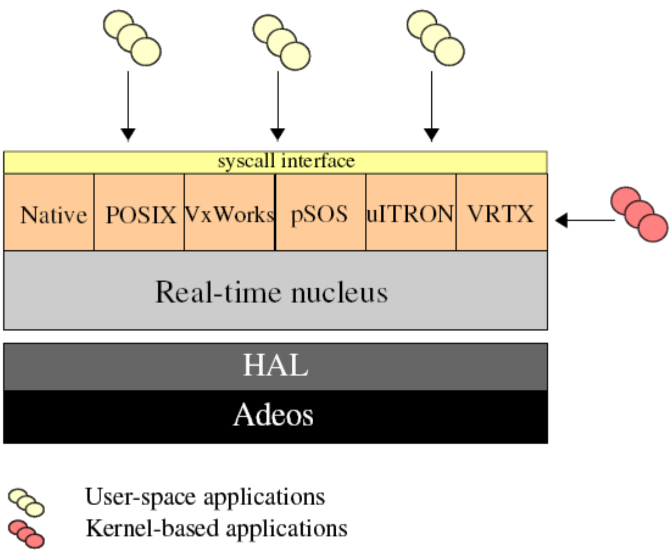


Figure 3.1: Architecture

The implementation of real-time tasks and communication with the core functions, occur using interfaces compatible with:

- VxWorks

- pSOS
- uITRON
- VRTX
- POSIX

For more information about the communication between processes is possible to consult the document Native-API-tour-rev-C.pdf, present on the site www.xenomai.org

## 3.4 Development Environment

### 3.4.1 IDE

The toolchain IDE is Eclipse. It takes care of starting the compilation and debug it. In Eclipse Git is also included.

The toolchain in a Microsoft Windows environment is released by Sirius.

### 3.4.2 Repository

For each board produced, Sirius keeps updated the operating system, checking periodically and incorporating bug fixes or proposed changes through customer feedback.

The management of all components of the operating system software (services, drivers, and utilities) is via the opkg package manager.

The server that contains the packages of the distribution (OPKG) for TAGBOARD is maintained by Sirius in open source form. This server also contains the Git repository on which it is published the history of the distribution and toolchain tags, from which the customer can collect the information about latest versions and choose whether to adopt them.

### 3.4.3 Debug

software architecture is so diveded:

- kernel space
- user space

To hide the complexity of the operating system the applications run in user space, with easily debuggable special real-time threads.

In the event of a system crash the kernel is able to indicate the point of failure to take appropriate decisions.

## 3.5 Update

Each operating system application (drivers included) or a specific application developed by or for a customer, is structured in packages.

This allows you to keep track of versions of each package and use the ethernet network to update the project under development.

## 3.6 Licenses

**kernel Linux** GPLv2

**librerie di sistema** LGPL

**applicazioni di sistema** GPLv2/GPLv3

**driver Sirius into kernel core** GPLv2

**driver Sirius out of kernel core** vary according to the agreements with customers

**applicazioni in user space** vary according to the agreements between customer and developer

**applicazioni in kernel space** GPLv2