MICROCHIP

Aerospace Marketing

CAN for Space

Eric TINLOT – ESA CAN WORSHOP 2017

Atmel

CAN for Space

- Can basics functionalities
- Can Frame description
- Can native securities for space
- Microchip / Atmel CAN for space

Can basics functionalities

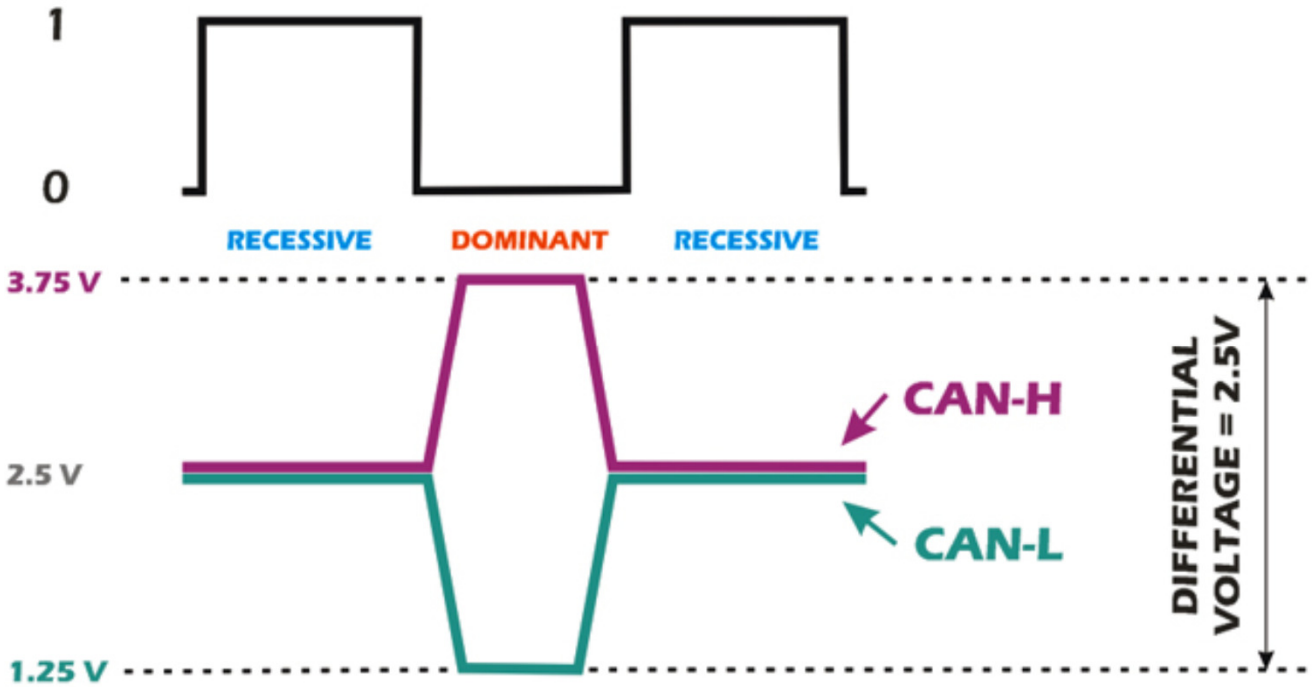
- The CAN bus is a multimaster serial bus communication system.
- When the network is not used, all node can transmit on the network.
- The Identifier can be on 11bits (CAN 2.0A) or on 29 bits (CAN 2.0B).
- An arbitration mechanism is implemented in the protocol using the Identifier.
  if two nodes transmit on the same time, the message with lowest ID will win the arbitration and be transmitted, the other one will wait until the end of the transmission to retry to transmit his message.)
- The number of node on a CAN network is not limited by the specification, typical value is 32 nodes. The maximum number of node is a mix between the node capacitance, the length of the network, and the bus speed.
- The length of a CAN network is depending of the bus speed.

Max Length (meters)	Bus speed
10	12Mbit/s (CAN FD)
40	1 Mbit/s
100	500 kbits/s
250	250 kbits/s

- Typical physical layer of a CAN bus is differential signals on twisted pairs to improve noise immunity.

Can basics functionalities

The bus state can be Dominant (logical 0) or Recessive (logical 1)

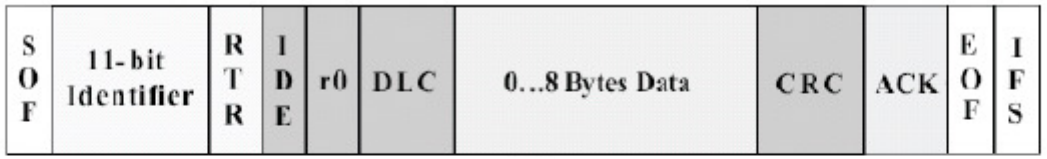


CAN for Space

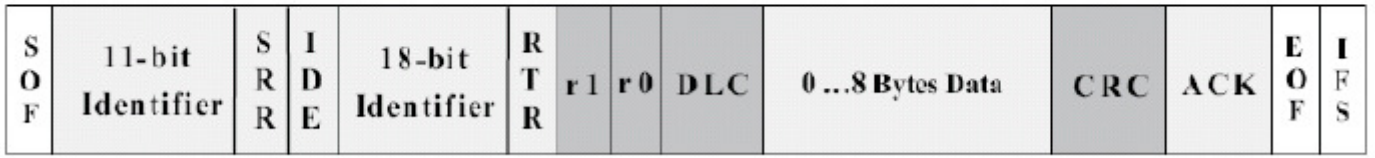
- Can basics functionalities
- Can Protocol description
- Can native securities for space
- Microchip / Atmel CAN for space

Can Frame description

CAN 2.0A



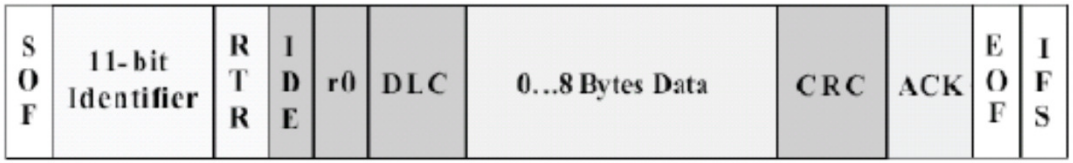
CAN 2.0B



SOF: Start of frame
SRR: Substitute remote request bit
IDE: Identifier extension
RTR: Remote Transmission request
R1/R0: Reserved bit
DLC: Data Lengh Code

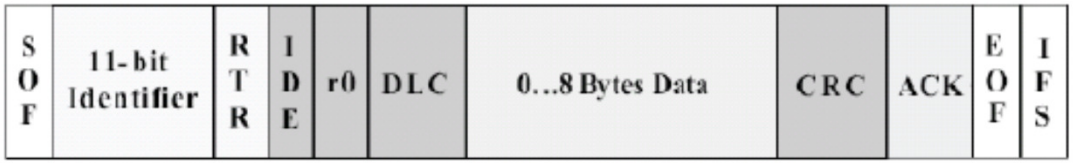
CRC: Cyclic Redundancy Check
ACK: Acknoledge field
EOF: End Of Frame field
IFS: Interframe Space

Can Frame description



- SOF - Start of Frame bit. It indicates start of message and used to synchronize the nodes on a bus. A dominant bit in the field marks the start of frame.
- IDENTIFIER - It serves dual purpose one, to determine which node has access to the bus and second to identify the type of message.
- RTR – Remote Transmission Request. It identifies whether it's a data frame or a remote frame. RTR is dominant when it is a data frame and recessive when it is a remote frame.
- IDE – Identifier Extension. It is used to specify the frame format. Dominant bit is for standard frame and recessive for extended frame.
- R0 - Reversed bit. Not used currently and kept for future use.
- DLC – Data Length Code. It is 4 bit data length code that contains the number of bytes being transmitted.

Can Frame description



- **DATA - Used to store up to 64 data bits of application data to be transmitted.**

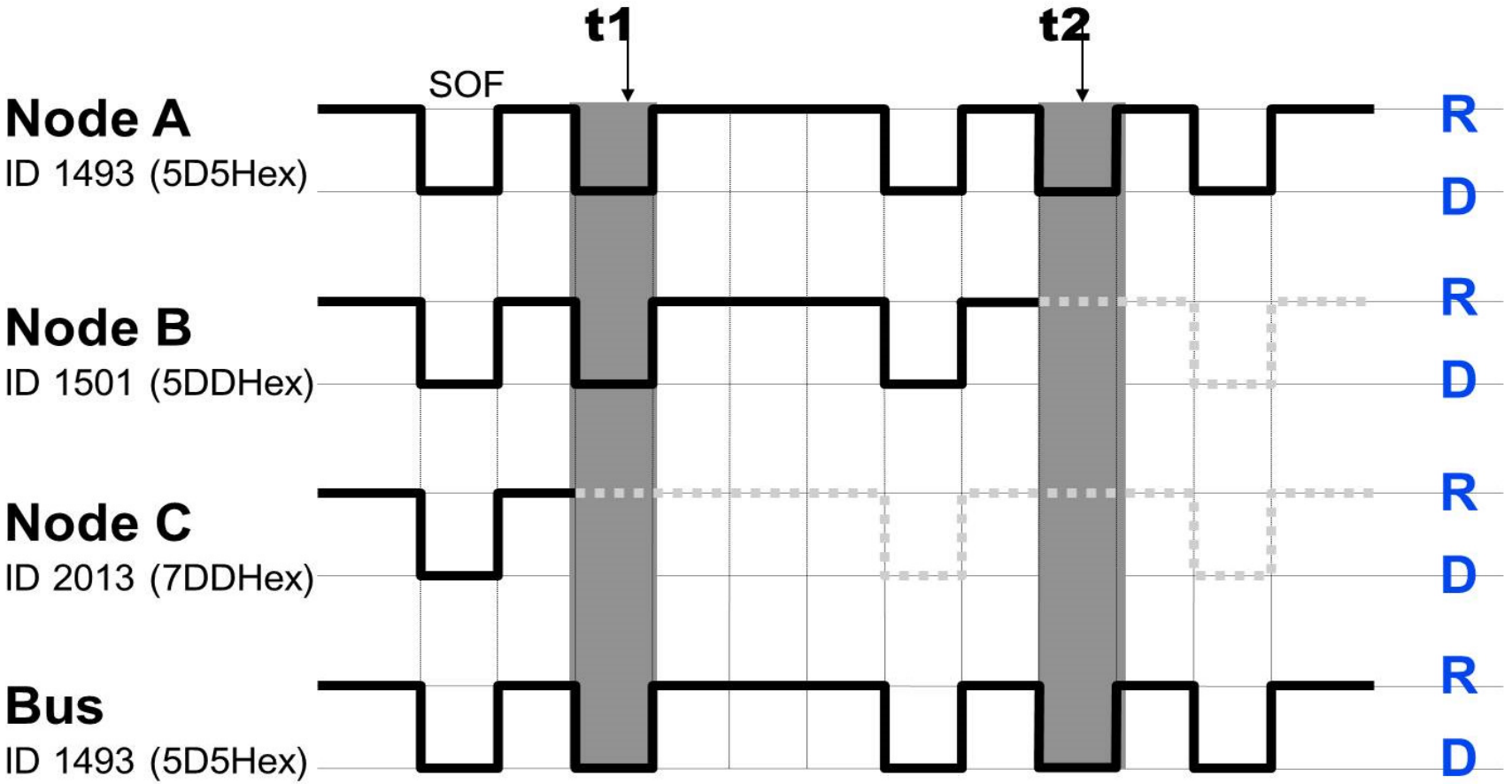
- **CRC - Cyclic Redundancy Check. The 16-bit (15 bits plus delimiter) cyclic redundancy check (CRC) contains the checksum of the preceding application data for error detection.**

- **ACK - Acknowledge (ACK) field. It compromises of the ACK slot and the ACK delimiter. When the data is received correctly the recessive bit in ACK slot is overwritten as dominant bit by the receiver.**

- **EOF - End of Frame (EOF). The 7-bit field marks the end of a CAN frame (recessive bits)**

- **IFS - Inter Frame Space that specifies minimum number of bits separating consecutive messages. It provides the intermission between two frames and consists of three recessive bits known as intermission bits. This time allows nodes for internal processing before the start of next frame.**

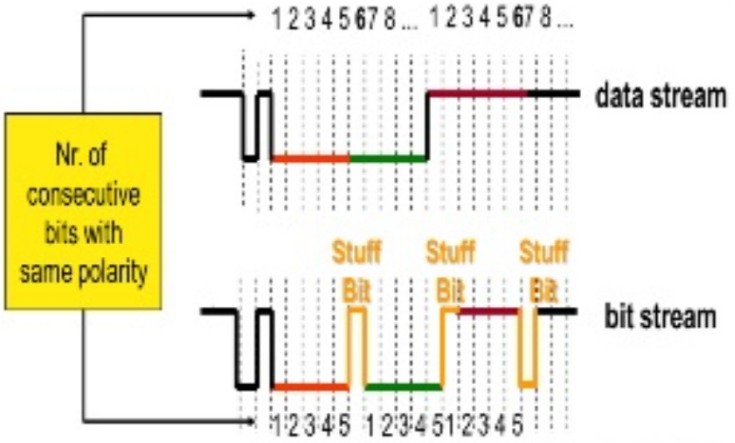
CAN Arbitration Example



t1 & t2 , Node C and Node B lost arbitration

CAN Bit Stuffing

- When five consecutive bits of the same polarity transmitted:
  - Transmitter insert one additional bit of the opposite polarity.
- The receiver checks the number of bits with the same polarity
  - Removes the stuff bits again from the bit stream. "destuffing".

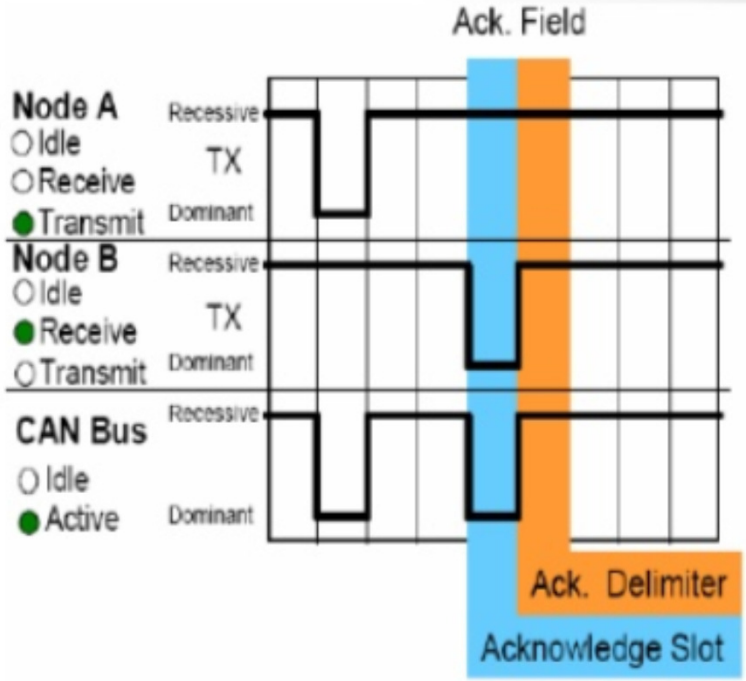


CAN Frame Acknoledge

CAN Error Detection:

Acknowledgment (ACK)

- A frame must be acknowledged by at least one other node.
- Otherwise ACK Error.



CAN for Space

- Can basics functionalities
- Can Frame description
- Can native securities for space
- Microchip / Atmel CAN for space

Bit Monitoring

The transmitter of a bit compares the signal sent with the signal seen on the bus line (transmission channel).

Except for the arbitration phase during the transmission of the identifier of a message and in the ACK slot, a transmitter starts sending an error frame if the signal on the bus line is different from the signal sent.

In this way, bit errors affecting all stations on the bus cannot lead to non-detectable errors because they will be detected by the transmitter of the frame.

CRC, Frame Check and Bit Stuffing

Cyclic Redundancy Check:

The 15 CRC bits are computed from every bit from SOF to the last data bit

Frame Check:

The SOF, RTR, IDE, DLC, delimiter and EOF fields must be consistent with the CAN specification. If a fixed format field in a received frame does not conform to the standard, the receiver sends an error frame and does not accept the received frame.

Bit Stuffing:

Any violation of the stuff rule between SOF and CRC is regarded as an error.

Acknowledgement:

The transmitter of a data or remote frame treats a missing acknowledgement as an error and destroys the EOF field by sending an error frame.

Error Counters

In addition to the above error detection mechanisms, each node maintains two error counters: the **Transmit Error Counter** & the **Receive Error Counter.**

There are several rules governing how these counters are incremented and/or decremented. In essence, a transmitter detecting a fault increments its Transmit Error Counter faster than the listening nodes will increment their Receive Error Counter.

The result of those counter is that a node who is always transmitting errors on the network (for whatever reasons) will after few retry of transmission automatically going into Bus-Off mode (disconnected from the network), and retry later on to connect on the bus or be switched to any redundant network.

All those securities on the protocol level allows to avoid any issue on the communication itself, and brings the CAN bus as a must for reliability in space communications.

CAN for Space

- Can basics functionalities
- Can Frame description
- Can native securities for space
- Microchip / Atmel CAN for space

Atmel Microchip Aerospace
Rad Tolerant Microcontrollers

- **SEL LET > 30 to 60Mev**
- **SEU LET 3 to 15Mev**
- **TID 20 to 50KRad**
- ***Space grade QML like***
- ***Hirel Plastic AQEC***

Flex Matrix w QoS

DSP Co-processing

QFP64 to QFP144

AES crypto

ECC

Ethernet

TCM

LIN

-55°C / +125°C

DAC

PWM

CAN

SERCOM

Motor control

ADC

SAMA5
Cortex-A5
NEON

850 DMIPS

SAMV71
Cortex-M7
+FPU

600 DMIPS
512KB to 2MB Flash

SAM4
Cortex-M4
+ FPU

180 DMIPS
512KB to 2MB Flash

SAM3XE
Cortex-M3

105 DMIPS
256KB to 512KB Flash

SAMD21
SAMC21
Cortex-M0+

45 DMIPS
8KB to 256KB Flash

ATmegaS
128/64M1
AVR8

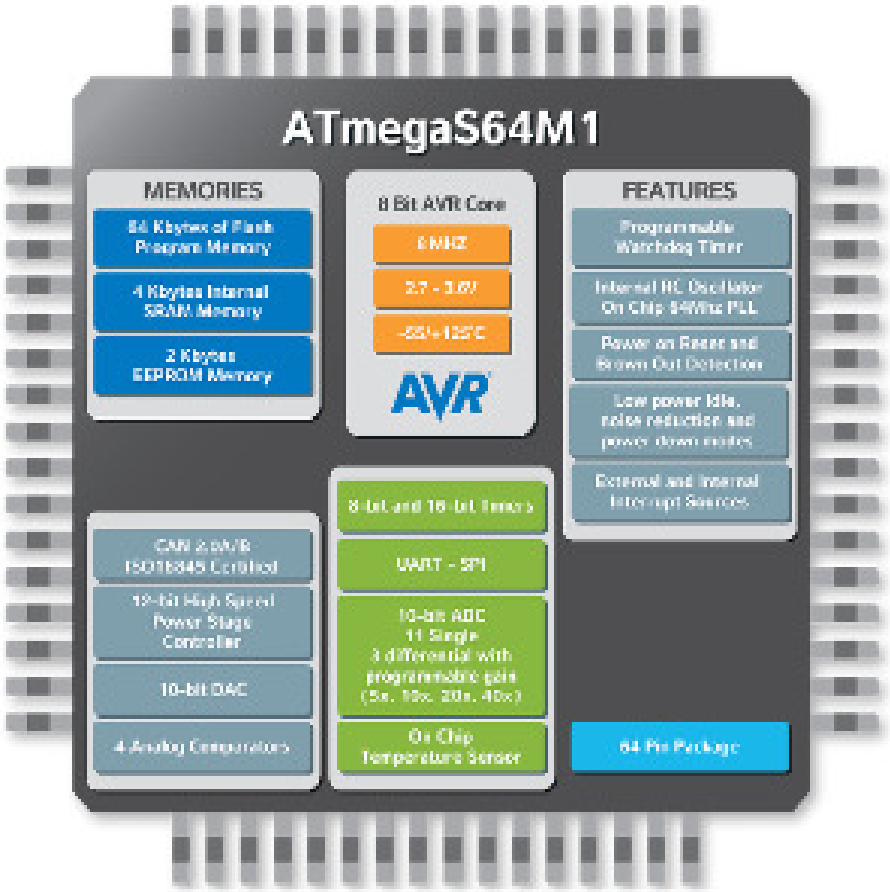
10 MIPS
512B to 384KB Flash

SERCOM = Serial Communication Module, Configurable UART, TWI, SPI,…

RT development ongoing

✔ FM available

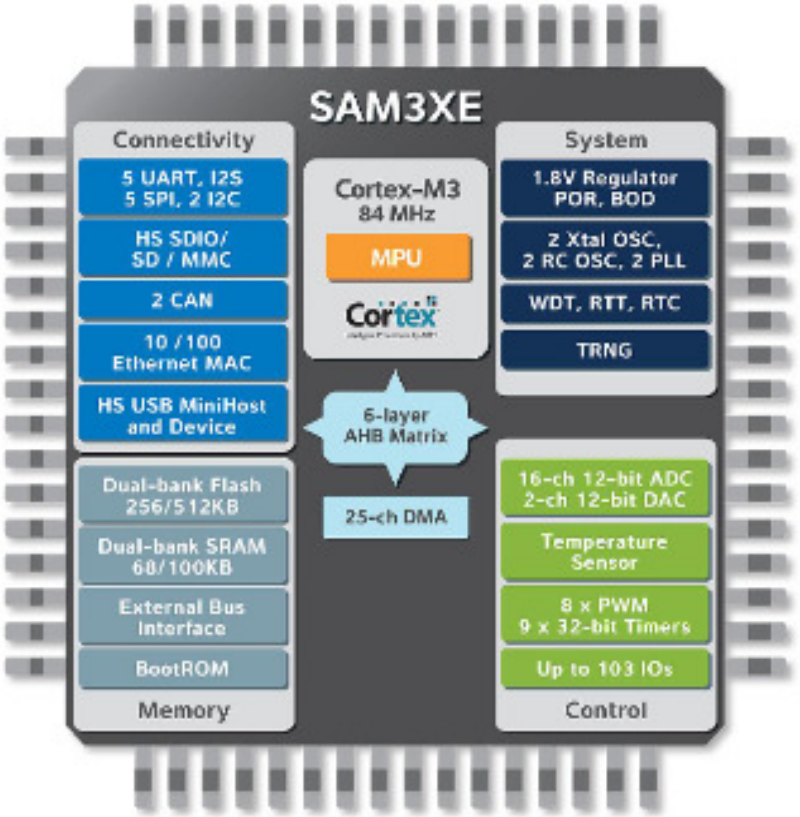
ATmegaS64M1



ATmegaS64M1 : a 8-Bit radiation tolerant AVR core microcontroller with **one CAN controller CAN2.0A&B** including 6 message objects - ISO 16845 certified.

Ceramic QFP32 pins
Plastic QFP32 pins

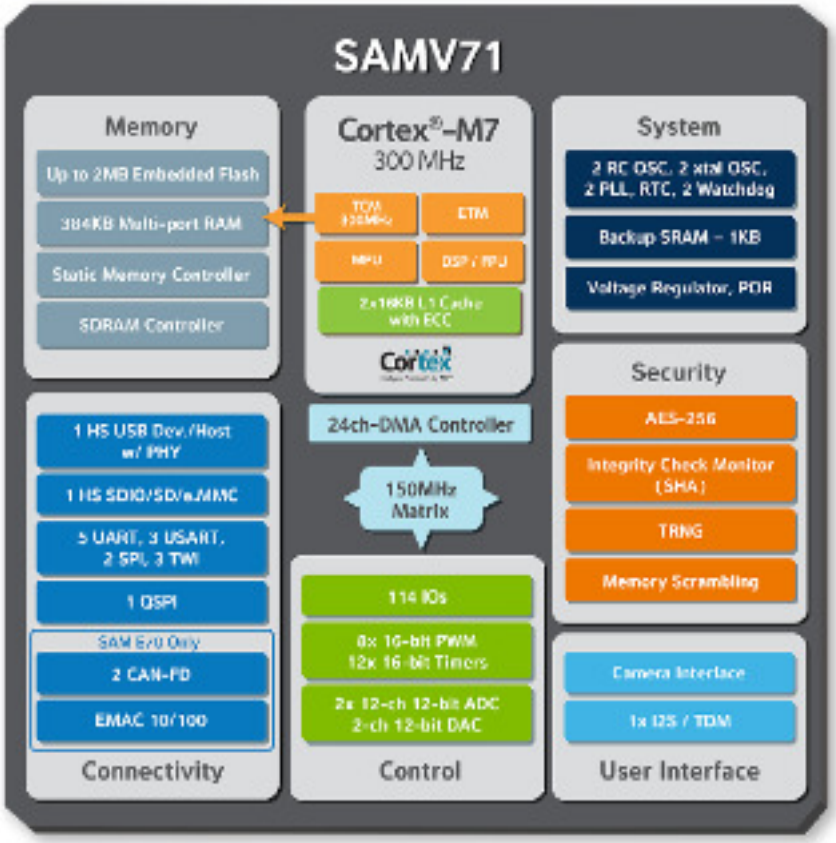
SAM3XE



SAM3XE: a radiation tolerant 32-Bit ARM Cortex-M3 that embeds **two CAN controllers CAN2.0A&B** - ISO 11898-1 certified, with 8 message objects.

Ceramic QFP144 pins
Plastic QFP144 pins

SAMV71



SAMV71: a radiation tolerant 32-Bit ARM Cortex-M7 that embed **two CAN FD controllers CAN2.0A&B** - ISO 11898-1 certified with up to 64 data bytes supported, up to 64 dedicated receive buffers and to 32 dedicated transmit buffers.

Ceramic QFP144 pins
Plastic QFP144 pins

CAN for Space w Microchip Atmel

- Benefit from state of the art technology
- Benefit from widely deployed solutions
- Benefit from easy access HW & SW
- Benefit from Space application expertise
- Not exposed to US export constraints

Welcome to our booth to learn more about our CAN solutions for Space