JED AVR256 standard architecture for ATmega2560 applications

The ATmega2560 is a standard ATmega core CPU with 8K RAM, 256K FLASH, 4K of EEPROM and is in a 100 pin package. It has 4 serial UARTs, I^2C and SPI interfaces, and 86 I/O lines. The full device data-sheet is 449 pages long, and is available at www.atmel.com.

The JED ATmega2560 board is an example of a CPU applications board for general use using this CPU, but differs dramatically from the usual board sold by microprocessor companies, in that all the inputs and outputs are industrial interfaces. **This is an industrial-strength board, useable in real-world applications:**

- Sixteen "Analog" ports, which can also be digital inputs, eight with pin-change interrupts. (These can, for example, handle 2-phase rotary position or shaft encoders very efficiently);
- Sixteen other ports can be digital in or out. Four also have "input capture" for waveform capture/analysis, and four feed counter/timer inputs. Outputs can have up to seven Pulse Width Modulation outputs;
- Output ports are 10 amp FETs with built-in high-energy protection and 60 volt flywheel clamp on the chip;
- Back-to-back clamp diodes are placed on all user lines for ESD protection;
- Series resistors for current and EMC limiting on all inputs;
- Voltage divider resistor pairs for setting input thresholds and voltage ratios and full-scale ranges for all analog and digital inputs. Analog input is 10-bit resolution by up to 16 channels. Basic input range is 0 … 4.096 volts;
- One additional port is a 1-Wire port for device expansion or operator certification/identification;
- Four RS232 Serial ports have ESD protection and EMC filters;
- Ultra-stable Real Time Clock with Temperature Compensated Crystal Oscillator (TXCO) for +/- 3.5 minutes/year max error. Clock device also has a 3 degree accuracy temperature sensor as board internal temperature readout;
- SPI and I^2C expansion ports to add-on upstairs boards to allow for more com ports, or high-resolution Delta-Sigma ADC (eg for Thermocouples or stain gauges for accurate load-weighing systems) or DAC (Analog output);
- Optional text or graphic LCD display and keyboard interface;
- Optional 2-axis digital accelerometer chip for shock detection;
- All devices used are full industrial temperature range and board is designed for lead-free manufacture.

Flexible memory expansion is provided: CPU RAM can be expanded to 64 Kbyte, a 32Kbye non-volatile FRAM and a 4/8 Mbyte DataFlash provide non-volatile data logging memory, and an optional SD/MMC interface can provide many Mbytes of removable data logging memory. Host USB connection is available to "USB Memory Sticks".

Four serial ports with flexible loading of interfaces for the four CPU UARTs is provided, so as well as **RS232** on all four ports, options allow for:

- USB slave port for communication to a PC (serial port emulation in the PC;
- USB host port, to communicate with memory sticks;
- uBlox LEA-4H or -4S GPS for high sensitivity, low current (38mA) positioning, able to accept DGPS corrections;
- 802.11 or other radio (eg 433Mhz or Zigbee) or interfaces to cellular phone networks or phone line modems;
- X-port for 10/100 Ethernet interface for networking or web page hosting on-board the AVR256;
- RS485 networking from the AVR256 to interface to off-board local data gathering/control networks.

These facilities make the AVR256 board ideal for asset, vehicle or mobile machinery monitoring and tracking.

Flexible power control compatible with 12 volt vehicle or solar powered and battery powered systems is provided, allowing systems to auto-start from a real-time-clock alarm, or from an external event such as an ignition turn-on in a vehicle. (Customised versions of this board could have protection and switching regulators for 24 volt vehicle environments.) Communication devices and ports can be powered up and down under program control.

This "S**tandard Architecture AVR256 board**" allows a list of I/O devices to be interfaced consistently across many projects. The standard board layout is useable directly in many cases where no custom pcb size or mountings is called for. However, if customers need a particularly compact design, the track-work on the pcb can be re-laid out, with maybe some different external interfaces (eg relays or opto-isolators or amplifiers) and connectors. If the port addresses are preserved in the re-layout, then the drivers will run unaltered on the custom board, but software development can start within minutes of powering up this standard board and then transferred to the customised or "cut-down" version.

Digital Inputs, Analog Inputs and FET outputs

The large I/O count of the CPU allows this board to have many (36) lines directly allocated to user I/O screw terminals (and Molex/Panduit pin headers) as user I/O, providing a wide range of analog input (16 lines) interrupt inputs (10 lines), timer inputs (5 lines), and (8) PWM outputs to FETs as drives for stepper or DC motors, and a 1-Wire port. The general-purpose I/O lines can have all 36 as analog or digital inputs, or up to 32 of these can be power FET outputs. Four-way DIP switches or LEDs can also be loaded in these positions or can connect via the screw terminals.

The sixteen "analog" inputs all can have user-installed pull-down resistors (eg 200 Ohms to terminate 4-20 mA current sources), can have user-installed pull-up resistors to Vcc to provide a top resistor and hence, current to a resistive transducer with one end to ground, a user-installed series resistor for protection, and then an optional second pull-down resistor as the bottom end of the divider pair (working with the series resistor), to scale higher input voltage input ranges to a range of 0 to 4.096 volts. The analog range is 0 to 4.092 volts for a count of 0-3FFh. An accurate, low drift MAX6033, 4.096 volt reference is used. ADC conversion is a successive approximation converter with 10 bits of resolution. Speed, noise and power control registers are provided, and channel gains and differential configurations combining channels can also be selected. (The ATmega2560 data sheet has 20 pages covering the ADC system!)

All pins of these "input" pin group can be used for digital inputs (read in parallel as CPU ports PF0 … PF7 and PK0 … PK7.) The PK0 … PK7 group are also "pin change" interrupt inputs which can power up a sleeping system, and/or generate an interrupt on the rising or falling edge, or a "low" level. (Three other interrupts are available on other I/O group terminals). Alternatively, FETs can be loaded and these can become four groups of four general-purpose outputs. (One interesting application for the "pin change" interrupt group is to process two-phase shaft-encoder inputs. Two interrupts are allocated for A and B inputs, and, when one pin interrupts, the other pin is read to determine direction of rotation. Thus these eight Port K lines can process four shaft encoders.)

The second group of 16 multi-purpose user I/O lines is for digital I/O. They are allocated from a range of CPU pins, and can be loaded with FETs to become outputs, or pull-up or pull-down resistors, series resistors and voltage divider pairs, when used as inputs with user-settable thresholds.

As outputs, eight ports are able to drive FETs from adjacent port pins (PE2 … PE5 and PH3 … PH6), which could be used to drive two groups of parallel outputs with identical timing, for example two four-phase stepper-motors. They can be used as four or eight more general-purpose inputs as well.

As inputs, another group of four pins are scattered around CPU ports, but are allocated to pins with "Input Compare" functions ICP1, ICP3, ICP4 and ICP5. Alternatively, FETs can be loaded and these can become four more general-purpose outputs.

As inputs, another group of four pins are scattered around CPU ports, but are allocated to pins with "Timer Input" functions CT0-In, CT1-In, CT4-In, and CT5-In. (CT3-In is also available on the SD/MMC alternate pins.) Alternatively, FETs can be loaded and these can become four more general-purpose outputs.

While the AVR256 board is mainly a surface-mount-technology board, the resistors, FETs and connector components associated with the 32 user I/O lines are all through-hole technology, so that JED or users can hand-load custom configuration resistors/FETs/connectors to tailor the board to an exact project requirement. Resistors can be plug-in into machine-pin socket strips. (FETs must be soldered in for mechanical stability and current-carrying capacity.)

Low value or zero-ohm resistors can be installed as series elements in all thirty-two I/O lines above, so a port can be available to external devices or interfaces essentially as an unencumbered CPU port pin. It could drive other CMOS or TTL logic or with current limiting resistors, drive either on-board LEDs or off-board LEDs or opto-isolators or opto-power relays. Small slider-switches can be installed in groups of four (with pull-up resistors) allowing users to use port bits for field-option-setting switches.

Three more input lines are available if the SD/MMC card option is not installed. These are lines from ports PB0(PCINT0), PE6(CT3-In/INT6) and PG3. They have SMT 10K pull-ups to 5 volts, series resistors and clamp diodes.

Another port pin with a 1K5 pullup and overvoltage ESD protection in available for Dallas 1-Wire use (PD5). It can also be a general-purpose user CMOS/TTL I/O logic line or contact closure input.

LCD display and Keyboard, I/O expansion

Another 19 CPU I/O lines (port PA, PC and PG 0 … 2) are pinned out to the LCD and keyboard interface connectors via ESD/EMC protection and 4K7 pullups to Vcc, to provide a flexible LCD/keyboard interface (alternating with extended 64Kbyte of SRAM). The LCD can be a 4 by 20, a 2 by 40 or a 4 by 40 text display (using two enable lines).

The keyboard can be a 4 by 6 or a 5 by 5 matrix or 10 individual switches to a ground pin on the keyboard connector.

It is also possible to use the combination of the "LCD" interface and the "keyboard" connectors to drive a graphics display. This interface is compatible with displays up to a 128 by 240 pixel format and a T6963C Toshiba controller. Software support for this display from an AVR CPU is included in several commercial compilers. (Four pins are still spare for key inputs.)

A flexible backlight powering system is provided, which allows the CPU to control the power on/off for the LCD backlight. An N-channel power FET is controlled by port PG0, and this grounds the –ve end of the backlight LED array. The positive end of the LED array can be fed from either a power resistor from the filtered input supply or a small DC/DC converter running from the filtered input supply when the FET is turned "ON" with a small series resistor for current limiting. This is the most efficient system with minimal resistive loss, complete CPU control and zero standby power. (It can also be fed from the 5 volt CPU supply in the mode when a DC/DC converter is used for CPU power.)

These ports can also be used for user I/O and pass through pull-up resistor and ESD protection and EMC filter devices. They can thus be used as CMOS/TTL level I/O with 19 logic lines (two full 8-bit buses from ports PA and PC plus three strobes) or 18 logic lines and power N-FET. EMC Filtered Vcc and Ground pins are also available on both connectors.

(If port PA, PC and PG 0 … 2 are used for CPU RAM expansion, keyboard and LCD functions can be connected to the CPU via the I^2C expansion connector using boards in the JED AVR200 family.)

SPI expansion system and CPU FLASH programming

The ATmega2560 CPU includes a powerful hardware implementation of "SPI", a high-speed clocked synchronous communications protocol (also compatible also with National's Microwire). Lines MOSI (Master Out Slave In), SCK (clock) and MISO (Master In Slave Out) use in turn CPU port bits PB2, PB1 and PB3. Each device communicating with the CPU via SPI needs a CPU pin allocated as a LOW true chip select. Port pin PB0 is used as the CS* pin for the MMC/SD memory card (see below).

The SPI system on this board is also used to communicate with the on-board DataFlash memory (using port PG5 as the low-true chip select) and the optional ADIS16006CCCZ accelerometer (using port PH2 as the low-true chip select.)

The SPI system is used for CPU FLASH memory programming, and a 14-pin connector on the board has the first six pins allocated to device programming, in a way compatible with Atmel's ISP programmer's 6-pin ribbon cable connector. (The ATmega2560 CPU does NOT use the awkward TX0/RX0 pins for ISP that the ATmega128 used.) Ground, 5volt Vcc and RESET* lines are on this connector as well. (See: http://www.atmel.com/dyn/products/tools_card.asp?family_id=607&family_name=AVR+8%2DBit+RISC+&tool_id=3808)

After a 4-pin gap for the programming header sides, there are another four pins from CPU port pins PB7 … PB4. These can be used for an off-board SPI-interfaced expansion board connected via a fourteen-way ribbon cable, or an above-board expansion board. The PB7 … PB4 pins are also CPU PCINT pins 7 … 4, so can be used as any combination of chip selects and CPU interrupt inputs. (There are a number of mounting holes on the AVR256 board to hold an upstairs board in place securely.)

If there is interest, JED could create a dual-channel UART-driven serial expansion board with the PB7 … PB4 pins used for chip, with two selects and two interrupts, one each for each channel, giving a total of six communications channels. Philips/NXP make this part.
(See: http://www.standardics.nxp.com/products/sc16/pdf/sc16is740.sc16is750.sc16is760.pdf)

Upstairs boards could also add extra functionality such a multi-channel voltage or current output Digital to Analog converter. There is also a family of 24 bit Delta-Sigma Analog to Digital converters from Linear Tech which could add a wide-range filtered micro-volt input signal direct conversion system for thermocouples, etc.)

Digital 2-axis accelerometer for shock sensing

An option on board for a 2 channel Analog Devices digital accelerometer (via the SPI interface) Part number is ADIS16006CCCZ. This can have user-installed damping capacitors to adjust system response.

(See: http://www.analog.com/UploadedFiles/Data_Sheets/ADIS16006.pdf)

I^2C expansion system

The ATmega2560 CPU includes a full hardware implementation of the I^2C communications system for expanding applications hardware to other chips, either on-board or off-board. (Atmel calls it the TWI or Two Wire Interface).

On the AVR256 board, the I^2C interface is used to communicate with the Real Time Clock (DS3232), and the FRAM non-volatile device (FM24C256).

Also the interface is available on a 10-pin header for expansion off-board or on an upstairs board. This header includes two 5volt Vcc pins and four ground pins, spaced between signal lines. The interface is non-buffered, and the pinout is compatible with a range of I^2C expansion boards designed by JED in connection with the AVR200 project board. (see: http://www.jedmicro.com.au/avr200.htm) Interface boards available include eight FET outputs, eight logic inputs, an LCD interface to a range of displays, and a 4 x 4 keyboard scanner. (If the 64 Kbyte memory option is installed in place of the AVR256 on-board LCD display and keyboard via Ports PA and PC, these functions can be added via a 10-way ribbon cable to these I^2C peripherals.) Upstairs boards could also add extra functionality such a multi-channel voltage or current output Digital to Analog converter.

A small buffer board plugs in above the AVR256 board, into the I^2C header to bi-directionally buffer and protect the I^2C lines and pin this out in the same way to allow long ribbon cables to communicate with multiple I^2C cables over a distance to the AVR200 family cards with buffers installed. This includes EMC filters on all lines)

DS3232 Real Time Clock, Battery backed RAM and Temperature transducer

The DS3232 provides a Real-Time-Clock to the AVR256 system. It runs from the I^2C bus, and as well as very accurate time and calendar functions, it provides an alarm capable of automatically powering up a sleeping system, and a block of 236 bytes of RAM with no read or write delays or wear-out characteristics. The timekeeping functions and the RAM are battery backed by a lithium coin cell held in a secure battery holder. The 32Khz crystal is inside the moulded package, protecting it from mechanical damage and effects from electrical or mechanical noise to the high-impedance crystal pins, which are now completely internal.

This RTC device uses a temperature-compensated crystal oscillator (TXCO) to provide a time-keeping accuracy of +/- 2ppm from 0 to 40 degrees C and +/- 3.5ppm from –40 to +85 degrees C. (This later is equivalent to +/- 2 minutes per year.) The clock counts from seconds to years and includes leap-year compensation to year 2099. There are two alarm register sets with time and date comparators. Enable registers enable one or both alarm systems to drive the interrupt output pin. This can be used to either power up a completely powered-down system via power supply control logic, or generate a PIN-CHANGE interrupt to a power-up CPU which is in idle mode.

The local temperature of the RTC chip can be read out of a register in the RTC. This is effectively the PCB temperature, as the RTC power dissipation is close to zero. The temperature accuracy is +/- 3 degrees Centigrade. Data is read out as a 9 bit number plus sign, with 0.25 degrees Centigrade resolution.

(See: http://datasheets.maxim-ic.com/en/ds/DS3232.pdf)

Expansion memory: CPU RAM

The lines used for the LCD and keyboard (above, port pins PA, PC and PG 0 … 2) can be loaded differently with an address latch, latching addresses from PA, strobed by ALE (PG2). This high-speed SRAM extends the working CPU RAM to 64Kbytes total. (Normally the CPU has 8 Kbyte of internal RAM.)

Expansion memory: non-volatile data storage

A flexible range of non-volatile data logging or file-providing memory is provided on the AVR256 card:

- An Atmel DataFlash (4/8/16 Mbyte capacity) chip accessed via the SPI bus provides high-speed sector-erased data storage with sector-sized dual ram buffering on the chip. Port PG5 is the DataFlash chip select (low true). (Two chip locations are provided for supply option flexibility between CASON and TSOP packages in 4 or 8 Mbyte capacities, but if both types are installed, and a CS* pin linked from a spare CPU pin, up to 16Mbyte of DataFlash can be loaded on the board.)
  (See http://www.atmel.com/products/DataFlash/ for data on the DataFlash devices);

- An FM24C256 or FM24C512, 32/64 Kbyte Ramtron FRAM (Ferroelectric Nonvolatile RAM (see (via I^2C), is provided for data buffering and non-volatile short-term storage during data logging. It has a 10 billion read/write cycle life (compared with 100,000 operations typical for EEPROM) and does not have the 10 ms erase and write times of an EEPROM. Typical applications are for full sector buffers for data being assembled prior to writing to DataFlash or USB memory stick. Data is generated by events as small records, and then written when a full sector has accumulated in the FRAM. This has speed and power consumption advantages in applications where data is needed in a steady stream, or needs to be logged with minimum power consumption.
  (See: http://www.ramtron.com/lib/literature/datasheets/FM24C256ds_r3.1.pdf) The FRAM devices are an interesting technology. (See: http://www.ramtron.com/doc/AboutFRAM/Default.asp);

- The DS3232 RTC provides 236 bytes of battery-backed RAM with no read or write delays or wear-out characteristics. This is accessible via the I^2C interface, and is useful for memory pointers, "last position" indicators, "run hours", event counters or "last authorisation logon" codes, ie variables which would cause wear-out problems if they were stored a static location in a FLASH memory;

- An SD/MMC card interface is an optional board-loading on the AVR256, and must be installed by JED. A 4-terminal Phoenix connector is replaced with a surface mount SD/MMC card socket mounted on one of the ends of the PCB, in such a position that slide-in cards could be accessible via a slot in a case or cabinet. If the physical location of the socket at the end of the PCB is not convenient, pads for a 10-pin IDC ribbon cable connector are provided between the SD/MMC card connector and the card edge, and this can be through-hole soldered into the board and it can then go to a panel-mounted SD/MMC connector off-board on a front panel, for example.

  The interface to the SD/MMC card is via voltage translation logic from the SPI bus, and CPU port pin PB0 is used as the SD/MMC card chip select. Ports PE6 and PG3 are used for "Card Detect" and "Write Protect" logic sensing.

  This interface is compatible with SD/MMC card file handling software from two sources which JED is aware of. A Google search shows several others are available as well:

  - Priio (see https://www.priio.com/productcart/pc/viewPrd.asp?idcategory=10&idproduct=29) make a CodeVison C software suite sold in source form (available from Dontronics in Australia) but licence-free for production. (The code occupies about 27 Kbytes of CPU Flash space and 1.5Kbyte of RAM space.) A full implementation is FAT12 and FAT16 compatible, with file sizes up to 1 Gbyte, so files can be exchanged with a PC in DOS/Windows compatible form, so, for example, logged data can be saved as simple comma-separated numbers with a CR/LF at the end of each record, and read directly into an Excel spreadsheet; and

  - MikroPascal (see http://www.mikroe.com/en/compilers/#avr), a PASCAL compiler for AVR which has built-in SD/MMC card software with a PC file structure.

- A USB memory stick interface is available for logging/file transfer, via a GHI or a VDRIVE1 USB host system which can be mounted on a bracket accessible externally. A flying cable connects from the USB host card to the main CPU card. A serial port connects to the interface at CMOS levels, and both devices have simple ASCII or binary serial commands. Files can be created, opened, read or written and directories can be created or deleted. Files on the USB stick are compatible with FAT file systems for Windows/DOS systems.
  (See: http://www.ghielectronics.com/details.php?id=1&sid=3
  and http://www.vinculum.com/prd_vdrive1.html)

Power supply and power switching

In a vehicle-based application, the board runs from 12 volt ("always on") vehicle power, and is transient protected with a polyfuse and transorb for load-dump or reverse polarity and filtered with a diode-protected large filter capacitor. (This filtered power is available on a screw terminal to run off-board equipment, eg a modem or radio. The filtering can be extended off-board, or an external Li battery or NiMH battery backup with charger can also be linked in here.

Power can switched on by RTC alarm or external ignition signal input from the vehicle or RI (Ring detect) from COM0 from a modem plugged into the board. The board will run from as low as 6.5 volts, but is fully protected for load dump and transients or reverse voltages for 12 volt automotive applications. Regulator power with the LM9070 linear regulator is 250mA with 800mV dropout (plus a series diode). (See: http://www.national.com/pf/LM/LM9070.html)

If more 5 volt power is needed, eg to power buffered I^2C expansion, radios, or for sensors/motors/relays etc or if the system is running from 24 volts (with front-end protection if in a vehicle), or from a 24 volt industrial regulated supply, the LM9070 can be replaced with a Recom switching regulator with a 6.5 to 34 volt input range and 5 volts out at 0.5A or 1.0 A (18v max). In this case, the 5 volt main switching regulator can also power the LCD backlight, if used.
(See: for 0.5A http://www.recom-international.com/pdf/innoline/R-78xx-0.5.pdf or
and for 1A http://www.recom-international.com/pdf/innoline/R-78xx-1.0.pdf)

Vcc at 5.0 volts is available on a pair of screw terminals to outside devices (or this board could have no regulators, and be powered completely from external regulators or a battery-backed power system can feed into these terminals, in low power logger applications.)

Vcc at 3.3V volts is regulated on board with a small linear regulator to run the GPS, the DataFlash components and the SD/MMC memory card.

The power supply control and regulator components can be loaded in a several different standard ways:

- In a low-power application (eg a vehicle-based data logger or vehicle or asset tracker), the CPU is powered by a quiet, linear regulator for best analog performance, and the LM9070 regulator has built-in logic for On/Off control suitable for vehicle monitoring systems. It has a 12 volt "ignition" sense input which powers up a system when the vehicle key is turned. A second input to the regulator is a "keep-alive" input generated by CPU port pin PJ7. Once the CPU is started by the ignition sense input, the regulator is kept powered by the "keep-alive" pin.

  The CPU also monitors the "ignition" input via port pin PJ6: if the "ignition" is switched OFF, the CPU remains active via the "keep-alive" pin, and can generate a final data logging function (eg recording position from the gps or sending out a "powering down" message via a radio link). The CPU can then close itself down in an orderly way, closing files, etc. Quiescent power from 12v in shutdown is 60 microamps.

  The CPU can control the power to the radio/gsm/gprs/modem via CPU port pin PL7. Vin-switched is available on a screw terminal to off-board modems, but the board can also be loaded with a wide-input range (6.5v/34v to 3.3 volt DC/DC converter (running from the switched Vin) to power the on-board, optional 802.11 DP101 2.4 Ghz modem or an ad-on wireless system plugged into the Serial-0 TTL-level interface;

- In a battery-powered and/or solar-powered data logging system, the CPU is powered via the linear regulator and the user can either run the CPU all the time and power it down internally with the flexible power control register, or close the CPU down and have the RTC power it up at regular intervals.

  If the CPU power is left on, all the other logic on the board, including the RS232 communications interfaces, and even the uBlox GPS can be powered to a very low power standby mode. The CPU can then come out of standby when an interrupt comes from the RTC, any of the off-board pin-change interrupts, or a ring-in on the modem, do a data log and close down to low power mode again. (The RX0 RS232 buffer is active in standby, and the RX0 pin is also a Pin-change interrupt 8.)

  It is also possible to have the CPU power up from an off-board modem receiving a phone or gsm/gprs call. In this mode, the modem is powered continuously from the "Filtered Vin" rather than the "Switched Vin". An Intercell SAM gsm/gprs runs from 5 to 30 volts and draws only 5 mA standby power. The RIN line on Serial-0 is used to wake-up the power supply from an OFF state;

A loading option for this board is to replace the linear 5 volt regulator with another wide-input-range switcher step-down regulator, which, while it might degrade the ADC fine resolution, it provides either 1 Amp or 0.5 Amp of 5 volt

power. This is used for applications where current required from the system is more than the 150mA provided, eg when driving a large group of I^2C interface cards. (If this is installed, it also runs the LCD backlight system.) If this switching regulator option is installed, it is still possible to do power switching of a system from the RTC, from SER0 RI (Ring detect) and the Ignition input. Switching of modem power is not possible in this mode, but the modem is powered continuously, so that it can wake a powered-down system.

Communications interfaces

- COM0 is a communications interface; default is a full 9-line DTE RS232 port on an IDC10 connector. (all RS232 standard lines are supported, connected to the D9 via a crimp-on 1-1 cable as a DTE. The RI line can optionally cause a sleeping board to power up.) This can go an off-board GSM/GPRS cellular RF modem, (typically a "SAM" from InterCEL. See: www.intercel.com.au, or to a land-line telephone modem. A link-able on-board option for this port is a DPAC WLNB-AN-DP101 802.11b or g port for wireless networking running from the comms switched DC/DC converter power supply.
  (See: http://www.quatech.com/catalog/airborne_wirelessdeviceserver_modules_emb.php)

  A third option for this com port is an eight-pin SIL connector with com pins available to a radio at 5 v TTL/CMOS levels (TX/RX/CTS/RTC/CD) Power on this connector is provided both from the CPU 5V Vcc supply (unswitched) and power from the comms switched DC/DC switcher, which could be 5V or 3.3V at up to 1 Amp. This radio could be a 151Mhz, a 433Mhz, a 915Mhz or a 2.4 Ghz transceiver device;

- COM1 default is RS232 on an IDC10 connector, with TX/RX lines provided. It is linkable to DTE or DCE pinout on the D9 cable (Other RS232 lines are looped back). An optional upstairs board for this port holds a uBlox LEA-4H Supersense GPS, with an antenna connection to a powered external antenna.
  (See: http://www.u-blox.com/products/lea_4h.html).

  A second option on this com port is a Lantronix XPort Ethernet 10/100 module, running from the 3.3 volt switched supply. (Physically, this alternates with the 802.11 on COM0 above, soldered onto the base board.
  (See: http://www.lantronix.com/device-networking/embedded-device-servers/xport.html);

- COM2 default is RS232 on an IDC10 connector, with TX/RX and CTS/RTS lines provided. It is linkable to DTE or DCE pinout on the D9 cable (Other RS232 lines are looped back or can be jumpered to +9 volts.) An option for this port is a "upstairs" RS485 interface to screw terminals, where the RTS line controls the RS485 TX enable;

- COM3 default is RS232 on an IDC10 connector, with TX/RX and CTS/RTS lines provided. It is linkable to DTE or DCE pinout on the D9 cable (Other RS232 lines are looped back or can be jumpered to +9 volts.). An option for this port via links can be serial at CMOS levels to a "micro-USB" USB-B input from a PC for code updates or data communications with a PC.
  (See: http://www.dontronics.com/pdf/uUSB_DataSheet_rev2.pdf).

  This module uses a Silicon Laboratories CP2102 device.
  (See: http://www.silabs.com/public/documents/tpub_doc/dsheet/Microcontrollers/Interface/en/cp2102.pdf)

CPU FLASH programming/re-programming

The AVR256 is programmed via the SPI system, using MOSI, MISO, SCK and RESET* pins, which appear on a 6-pin header compatible with the Atmel's AVRISP MKII. This can program any part of the CPU FLASH memory, including the boot sectors. These are stocked by JED in Australia, and have a USB connection to the development PC. See: www.atmel.com/dyn/products/tools_card.asp?family_id=607&family_name=AVR+8%2DBit+RISC+&tool_id=3808

The CPU book block sector can be loaded with a loader program to enable this board's CPU FLASH to be reloaded from a section of the DataFlash memory on reboot, and as this memory can be loaded by the CPU over any communications or file transport system provided on the board, potentially the CPU could be re-programmed from:

- A file on a SD/MMC memory card;
- A file on a USM memory stick;
- A file transferred from a PC into the "micro-USB" connection from a USB interface on any PC;
- A file transferred via any RS232 serial port;
- A file transferred via the 10/100 Ethernet connection in to the XPort device;
- A file transferred via an 802.11 network and the DPAC WLNB-AN-DP101;

- A file transferred via a GSM/GPRS radio modem plugged into COM0; or
- A file transferred via phone connection and a land-line to a conventional modem.

In each case, the user CPU program sets up the re-program sequence, communicating with it's host or unloading a transferred file into the allocated DataFlash sectors (and checking it with a CRC checksum). Then CPU sets some flags in the CPU eeprom, and does a jump to the restart address. On startup in the boot loader, the system checks the eeprom flags, and if set, proceeds to erase the CPU FLASH memory and re-writes it from the file, again doing a CRC check. When re-loaded and checked, the CPU clears the eeprom flags, and restarts again, this time (as the flag is NOT set) the user program starts normally, "phones home" and tells the control centre a successful reload has occurred.

The user-written program would be responsible for transferring the file to the DataFlash nominated area in whatever way was convenient, but they would all use the same boot loader in the boot sector of the DataFlash, which JED would provide.

AVR programming languages.

Many different compilers are available for the AVR, and assembly language can even be used, via the free "Studio" software from Atmel's web site or supplied with the AVR ISP programmer on CD. Atmel list compiled language support from 16 different companies at: http://www.atmel.com/products/AVR/thirdparty.asp#compilers

At JED we have used the CodeVison C for many years, and highly recommend it for large-scale projects as a very good implementation of ANSI C with good embedded libraries and good "code-wizards" for AVR hardware support (I^2C, SPI, 1-Wire, LCD, etc). Priio support package for SD/MMC written in CodeVision C. See: http://www.hpinfotech.ro/

We have also used the BASCOM BASIC compiler from MCS Electronics, and found it a simpler system than the C, but easily picked up by beginners. Has built-in I^2C, SPI, UART, LCD, 1-Wire support, simulator, etc. See http://www.mcselec.com/index.php?option=com_content&task=view&id=14&Itemid=41

and the PASCAL compiler from mikroElectronica ... seems easy to pick up and use. Has SD/MMC support built in, as well as I^2C, SPI, 1-Wire etc. See: http://www.mikroe.com/en/compilers/#avr),

There is also the free "gnu avr-gcc" tool-set, including the WinAVR IDE. See: http://www.avrfreaks.net/index.php?module=FreaksTools&func=viewItem&item_id=145

There are also a lot of code support products reviewed at the very popular dedicated AVR site: http://www.avrfreaks.net/

Sharing drivers

The big advantage of a "standard architecture" is that all I/O devices appear in the same location in port I/O space on the CPU, from project to project, across many customers, so savings with common drivers, common diagnostics, common documentation, familiarity (both at JED and at customers with multiple projects) occur.

It also means the standard "full" board can be quickly customised with the I/O ports needed for an application, peripherals loaded (gps, XPort, etc) and then customer software generation can be begun on a pcb which can, in quite a few cases, be the final production "build".

Demonstrations of project feasibility can also be done at low cost and financial risk to customers or management.