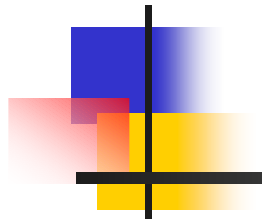


Ambiente di sviluppo per applicazioni custom 'Atomic'



ATOMIC
by **Ever**
ELETTRONICA

Introduzione

- ✓ 'Atomic' è un micro-linguaggio di programmazione sviluppato per la famiglia di azionamenti **'full digital'** di EVER Elettronica denominata SDM, MD, SW ed SM.
- ✓ Lo scopo di questo linguaggio è di offrire all'utente la libertà di creare la propria applicazione senza la necessità di passare ad azionamenti più costosi e complessi.
- ✓ La filosofia di 'Atomic' è di impiegare poche ma estremamente potenti macroistruzioni.
- ✓ Applicabile a tutti gli azionamenti della serie SDM, MD, SW ed SM con applicato il logo 'Atomic Ready' e configurati con il firmware 'Atomic' (c0499).

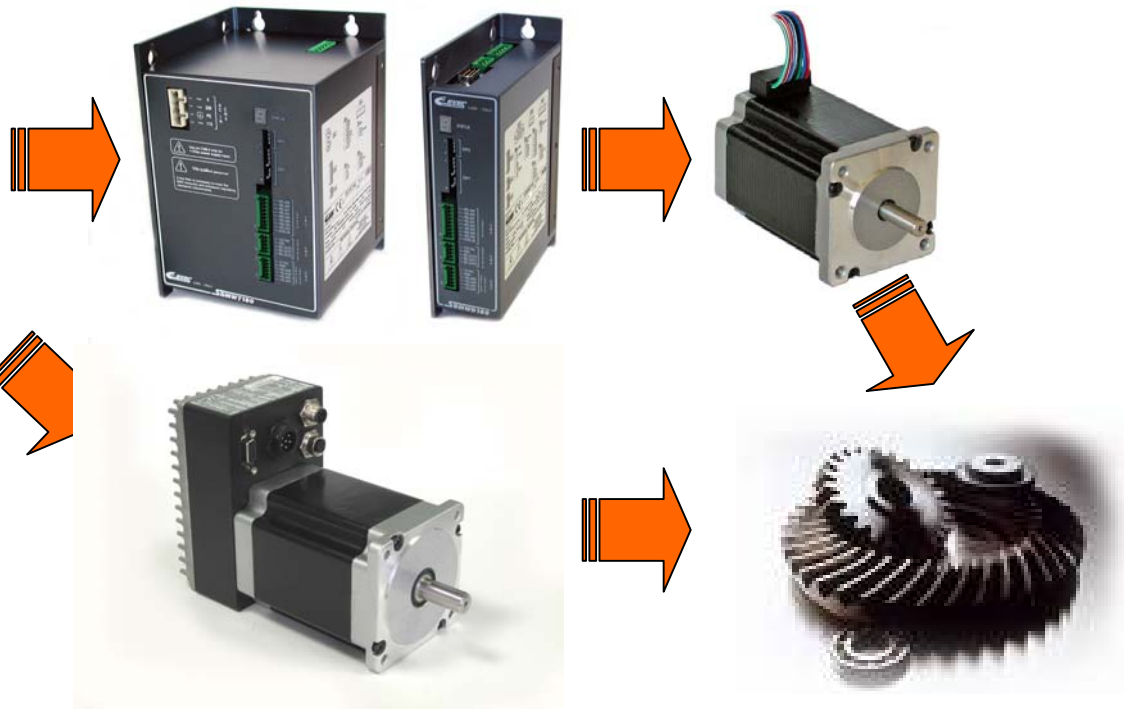


Introduzione

- ✓ La programmazione con 'Atomic' si effettua tramite una semplice interfaccia software per Personal Computer con sistema operativo windows, fornita dalla Ever Elettronica.



**Interfaccia
'Atomic'**



**'Azionamenti' SDM, MD, SW
ed SM con firmware 'Atomic'**

Componenti hardware e software

Tabella dei componenti hardware e software compatibili con 'Atomic':

Azionamento	Kit software
- SDMWD180vA133c0499 - SDMWD180vA143c0499 - SDMWA180vA133c0499 - SDMWT180vA133c0499	SDM_ATM-00
- SDMWD170vB231c0499 - SDMWD170vB242c0499 - SDMWA170vB231c0499 - SDMWA170vB242c0499 - SDMWA170v2231c0499 - SDMWA170v2242c0499 - SDMWA170v4231c0499 - SDMWA170v4242c0499	SDM_ATM-00
- SDMWA130vA136c0499	S130_ATM-00
- MDK3AxxxNvPA36c0499 - MDK3AxxxHvPA36c0499 - MDK3BxxxNvPA36c0499 - MDK3BxxxHvPA36c0499 - MDK3CxxxNvPA36c0499 - MDK3CxxxHvPA36c0499 - MDK3DxxxNvPA36c0499 - MDK3DxxxHvPA36c0499	MD_ATM-00
- SW1D4080N361-00c0499	SW_ATM-00
- SM2Ax60PNx43Ax0c0499 - SM2Ax60PNx43Bx0c0499 - SM2Ax60PNx43Dx0c0499 - SM2Ax60PNx43Ex0c0499	SM_ATM-00

I kits software comprendono:

- il **CD ATOMIC** per l'installazione dell'ambiente sul PC
- il **cavetto RS232** appropriato per l'azionamento utilizzato

Specifiche tecniche

✓ Queste sono le specifiche e le funzioni supportate dall'applicazione:

- 15 potenti macroistruzioni
- 8 applicazioni memorizzabili dall'utente
- Dimensione massima del programma 640 bytes
- Dimensione media della macroistruzione 6 bytes
- 32 variabili utilizzabili dall'utente
- Supporto di numeri interi a 4 bytes
- Accesso a tutti gli oggetti di controllo degli azionamenti
- Accesso a tutti gli ingressi/uscite degli azionamenti
- Supporto multitasking fino a 8 task utente
- <1 ms di tempo medio di esecuzione di un'istruzione

Vantaggi di Atomic

- ✓ Atomic introduce un nuovo modo di creare la propria applicazione:
 - Semplice, rapida e completa creazione del ciclo utente
 - Ampiamente versatile per ogni esigenza
 - Evita all'utente d'imparare la sintassi del linguaggio
 - Funzionante sugli azionamenti della serie SDM, MD, SW ed SM con firmware c0499
 - Integra tutti i vantaggi della tecnologia f^4d^2
 - Semplifica la diagnostica sull'applicazione
 - Sfrutta tutta la potenza dei DSP integrati negli azionamenti
 - Completo supporto e training da parte del personale di EVER Elettronica

Set di istruzioni

Istruzioni per la gestione del motore:

Istruzione	Descrizione	Parametri
MOVE ⁽¹⁾	Fa partire il motore in accordo con i parametri dati	1) Tipo di movimento
		2) [opzionale] Parametri di movimento (passi, posizione,...)
		3) [opzionale] Profilo di velocità
		4) [opzionale] Rampa di accelerazione
		5) [opzionale] Rampa di decelerazione
STOP ⁽¹⁾	Arresta il motore in accordo con i parametri dati	1) Tipo di arresto (nessuna rampa, con rampa, con passi,...)
		2) [opzionale] Parametri di arresto (numero dei passi,...)

(1) La funzionalità di queste istruzioni è identica a quelle integrate negli azionamenti con firmware CANopen std e Modbus std.

Set di istruzioni

Istruzioni con funzioni aritmetiche:

Istruzione	Descrizione	Parametri
ADD	<p>Esegue un'addizione tra due oggetti/variabili utente/costanti numeriche ed inserisce il risultato in un oggetto/variabile utente</p> <p>Destinazione = 1° operando + 2° operando</p>	1) Destinazione (variabile utente/oggetto)
		2) Primo operando (costante numerica/variabile utente/oggetto)
		3) Secondo operando (costante numerica/variabile utente/oggetto)
SUB	<p>Esegue una sottrazione tra due oggetti/variabili utente/costanti numeriche ed inserisce il risultato in un oggetto/variabile utente</p> <p>Destinazione = 1° operando - 2° operando</p>	1) Destinazione (variabile utente/oggetto)
		2) Primo operando (costante numerica/variabile utente/oggetto)
		3) Secondo operando (costante numerica/variabile utente/oggetto)
MUL	<p>Esegue una moltiplica tra due oggetti/variabili utente/costanti numeriche ed inserisce il risultato in un oggetto/variabile utente</p> <p>Destinazione = 1° operando * 2° operando</p>	1) Destinazione (variabile utente/oggetto)
		2) Primo operando (costante numerica/variabile utente/oggetto)
		3) Secondo operando (costante numerica/variabile utente/oggetto)
DIV	<p>Esegue una divisione tra due oggetti/variabili utente/costanti numeriche ed inserisce il risultato in un oggetto/variabile utente</p> <p>Destinazione = 1° operando / 2° operando</p>	1) Destinazione (variabile utente/oggetto)
		2) Primo operando (costante numerica/variabile utente/oggetto)
		3) Secondo operando (costante numerica/variabile utente/oggetto)

Set di istruzioni

Istruzioni di controllo del flusso dell'applicazione:

Istruzione	Descrizione	Parametri
WAIT	Arresta l'esecuzione dell'applicazione fino al verificarsi dell'evento specificato	1) Evento (tempo di ritardo, motore fermo o in movimento, apertura/chiusura ingresso). 2) [opzionale] tempo, numero dell'ingresso.
TEST	Cambia il puntatore della successiva istruzione in base al risultato del test	1) Tipo di test (apertura/chiusura ingresso, confronto (<>= !=) di due variabili utente od oggetti. 2) Variabile utente/oggetto. 3) [opzionale] Valore di comparazione. 4) Numero di linea se il test risulta vero.
JUMP	Cambia il puntatore della prossima istruzione	1) Numero di linea.
CONTEX_SWITCH	Passa ad un'altra applicazione residente nell'azionamento	1) Numero dell'applicazione.
END	Termina l'applicazione	Nessuna.

Set di istruzioni

Istruzioni di impostazione:

Istruzione	Descrizione	Parametri
SET	Imposta la destinazione uguale alla sorgente	1) Destinazione (variabile utente, oggetto, uscita digitale). 2) Sorgente (valore costante, variabile utente od oggetto).

Istruzioni booleane:

Istruzione	Descrizione	Parametri
BOOL	Esegue un'operazione booleana (AND, OR, SHIFT RIGHT, SHIFT LEFT)	1) Destinazione (variabile utente, oggetto). 2) Primo operando (variabile utente, oggetto). 3) Secondo operando (valore costante, variabile utente od oggetto).

Esecuzione dell'applicazione

All'accensione l'azionamento esegue:

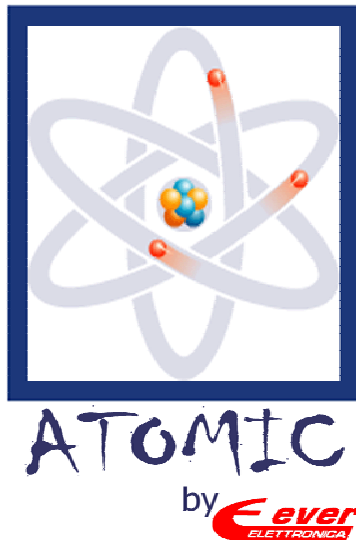
- il caricamento dei parametri di partenza (settaggi, corrente del motore, rampe, ...) residenti nella NVRAM, controllando se è residente anche una valida applicazione utente;
- l'esecuzione dell'applicazione utente dalla prima linea di istruzioni.

Durante la fase di funzionamento l'azionamento:

- **esegue** l'applicazione utente;
- **agisce** come uno standard slave CANopen/Modbus e tutti i controlli abilitati nell'oggetto 'Drive_Working_Settings' sono monitorati;
- **effettua** un monitoraggio dello stato di funzionamento del drive e se viene rilevata una condizione di emergenza hardware o software l'applicazione utente è arrestata sino a che permane la situazione di emergenza.

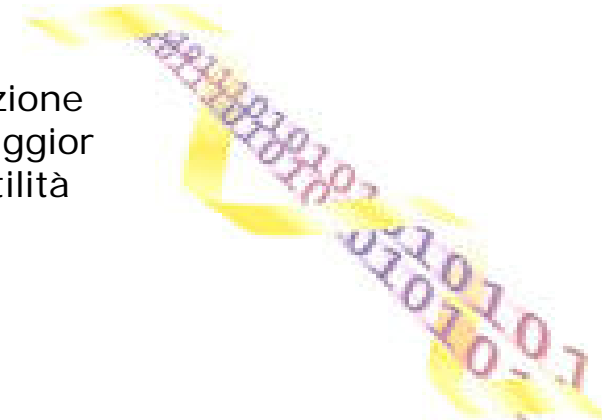
Introduzione alla sintassi

L'ambiente di sviluppo Atomic è un'applicazione per sistemi operativi Microsoft Windows^(tm) che permetterà all'utilizzatore di creare la propria applicazione per mezzo di wizards e di finestre di dialogo, senza alcuna necessità di scrivere nemmeno una sola riga di istruzioni.



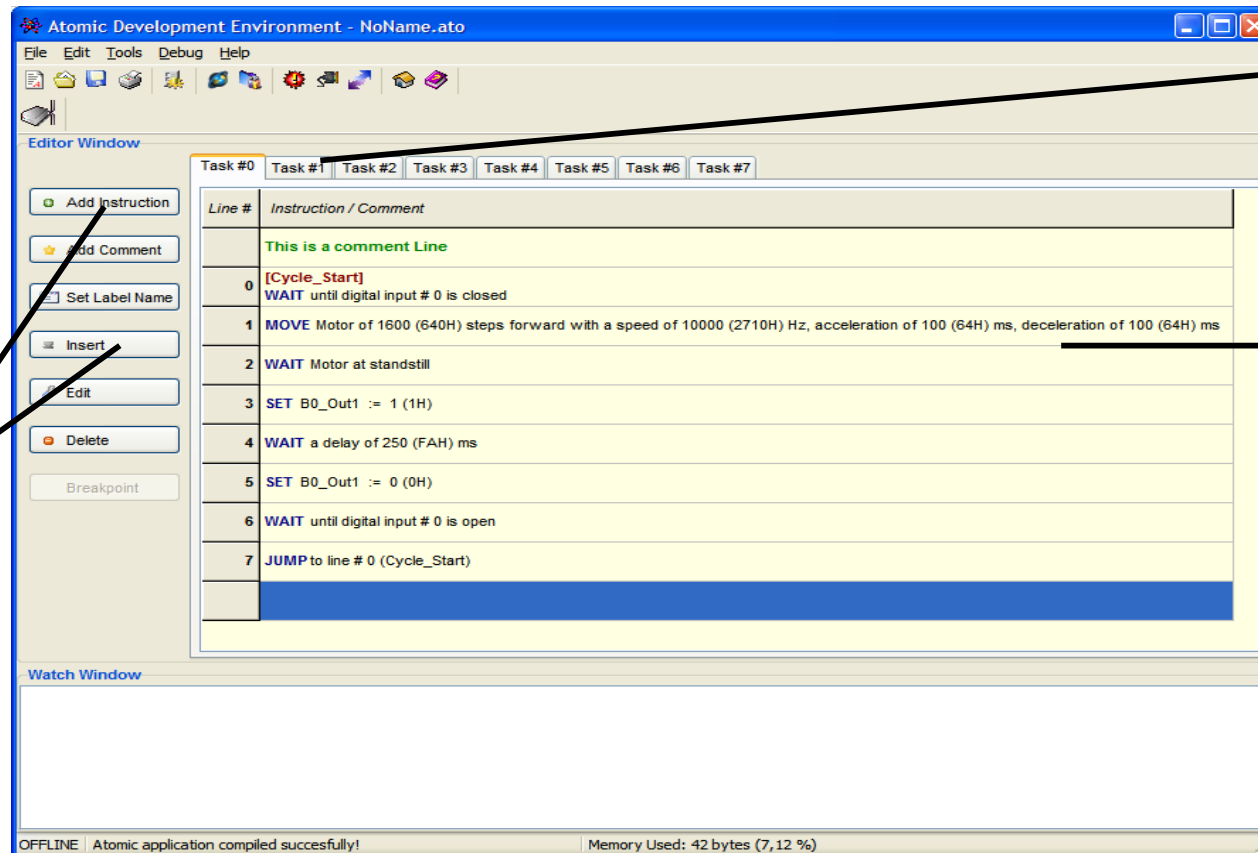
Ciò significa che l'utente, a differenza di altri linguaggi di programmazione, non dovrà imparare la sintassi di linguaggio delle macroistruzioni di Atomic che viene completamente gestita dal software installato sul Personal Computer.

Le varie finestre di configurazione sono studiate per trarre il maggior vantaggio in termini di versatilità e rapidità di compilazione.



Sintassi del linguaggio

Il programma utente creato in Atomic apparirà come nell'esempio seguente:



Tab dei Task utente

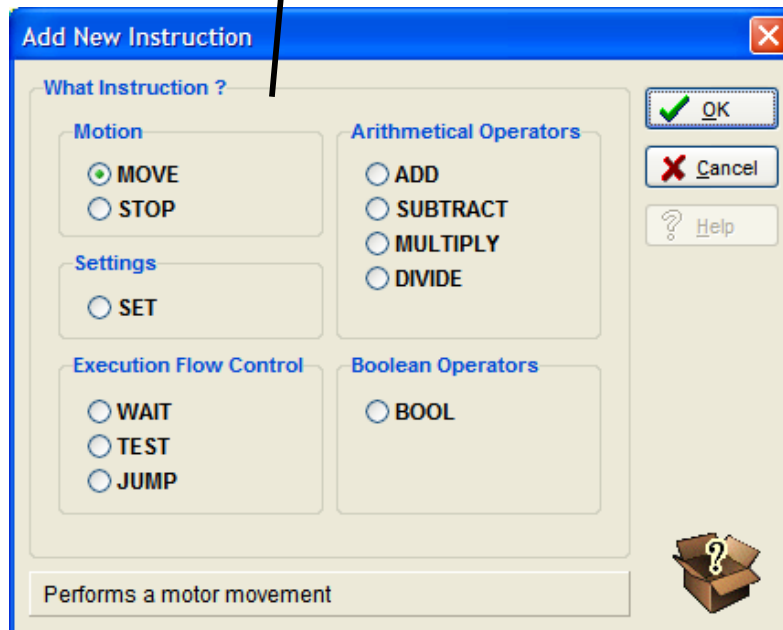
Area della sequenza delle istruzioni dell'applicazione utente

Pulsanti per aggiungere ed editare le istruzioni

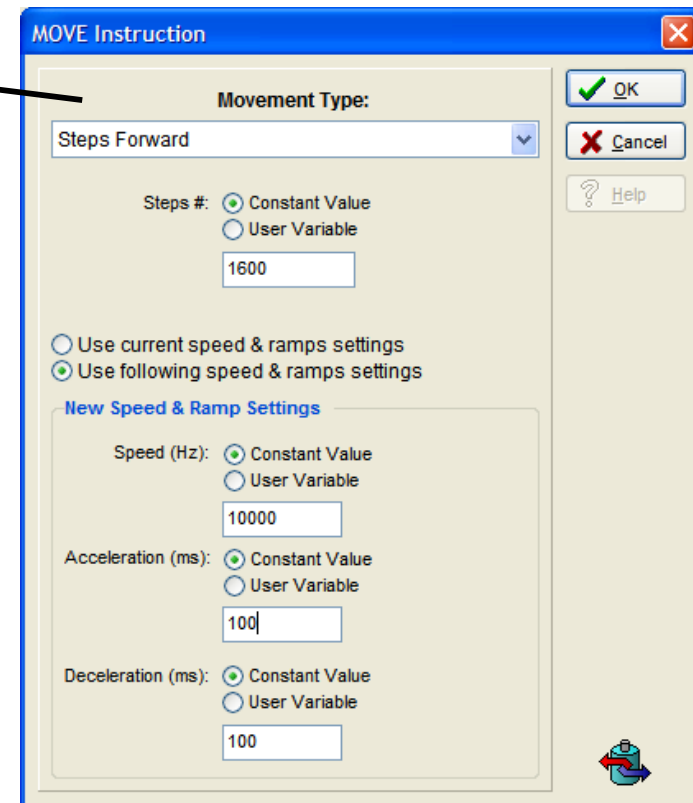
Sintassi del linguaggio

I pulsanti funzione per aggiungere ed editare le macroistruzioni aprono le seguenti finestre:

Per aggiungere una macroistruzione



Per settare i parametri di una nuova macro di tipo MOVE



Sintassi del linguaggio

I pulsanti funzione per aggiungere ed editare le macroistruzioni aprono le seguenti finestre:

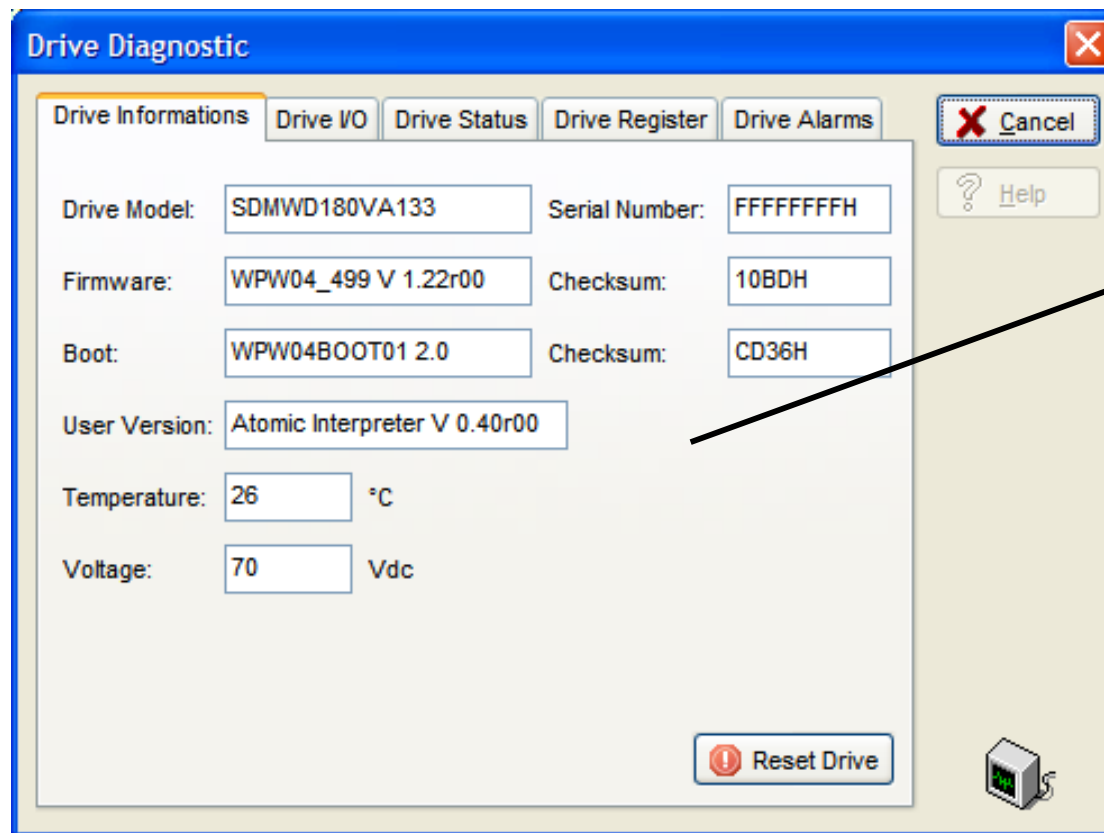
Per settare i parametri di una nuova macro di tipo SET

Per settare i parametri di una nuova macro di tipo BOOL

Per settare i parametri di una nuova macro di tipo WAIT

Diagnostica del sistema

Il pulsante per la diagnostica apre la seguente finestra:

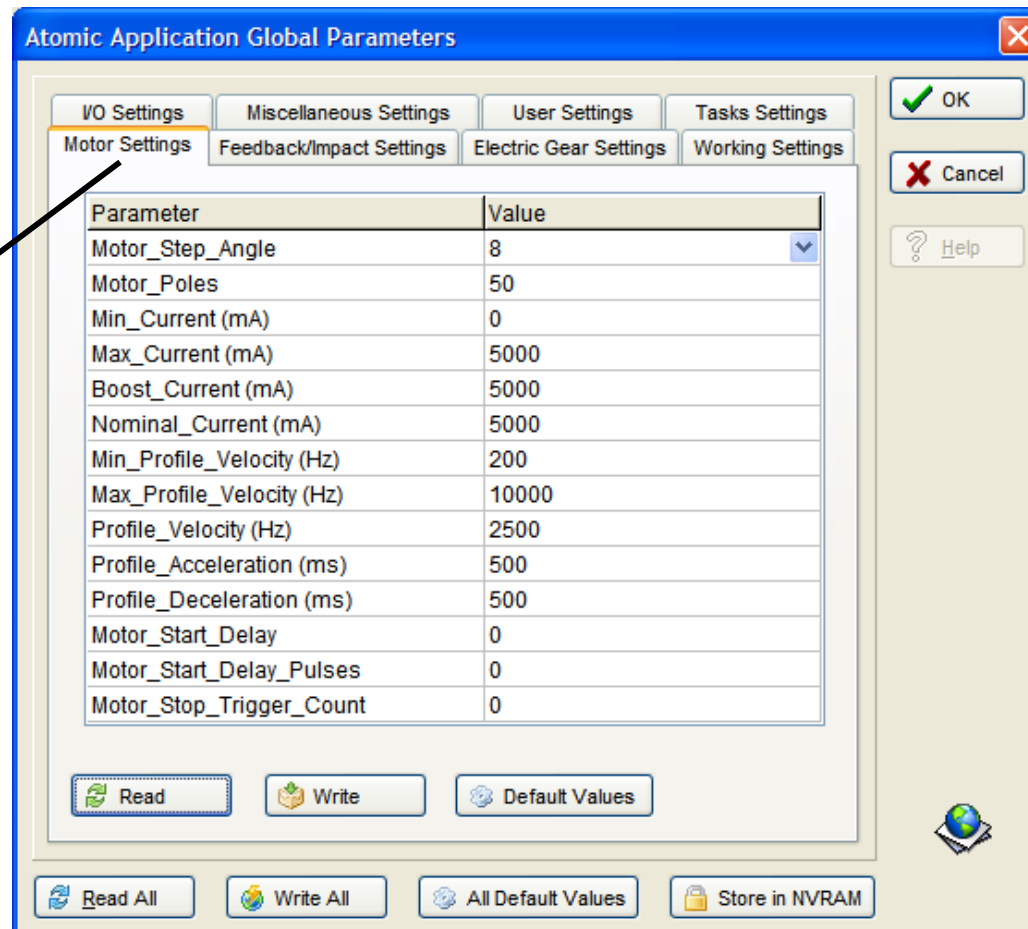


Field	Value
Drive Model	SDMWD180VA133
Serial Number	FFFFFFFFH
Firmware	WPW04_499 V 1.22r00
Checksum	10BDH
Boot	WPW04BOOT01 2.0
Checksum	CD36H
User Version	Atomic Interpreter V 0.40r00
Temperature	26 °C
Voltage	70 Vdc

Informazioni sul sistema in uso

Impostazioni di sistema

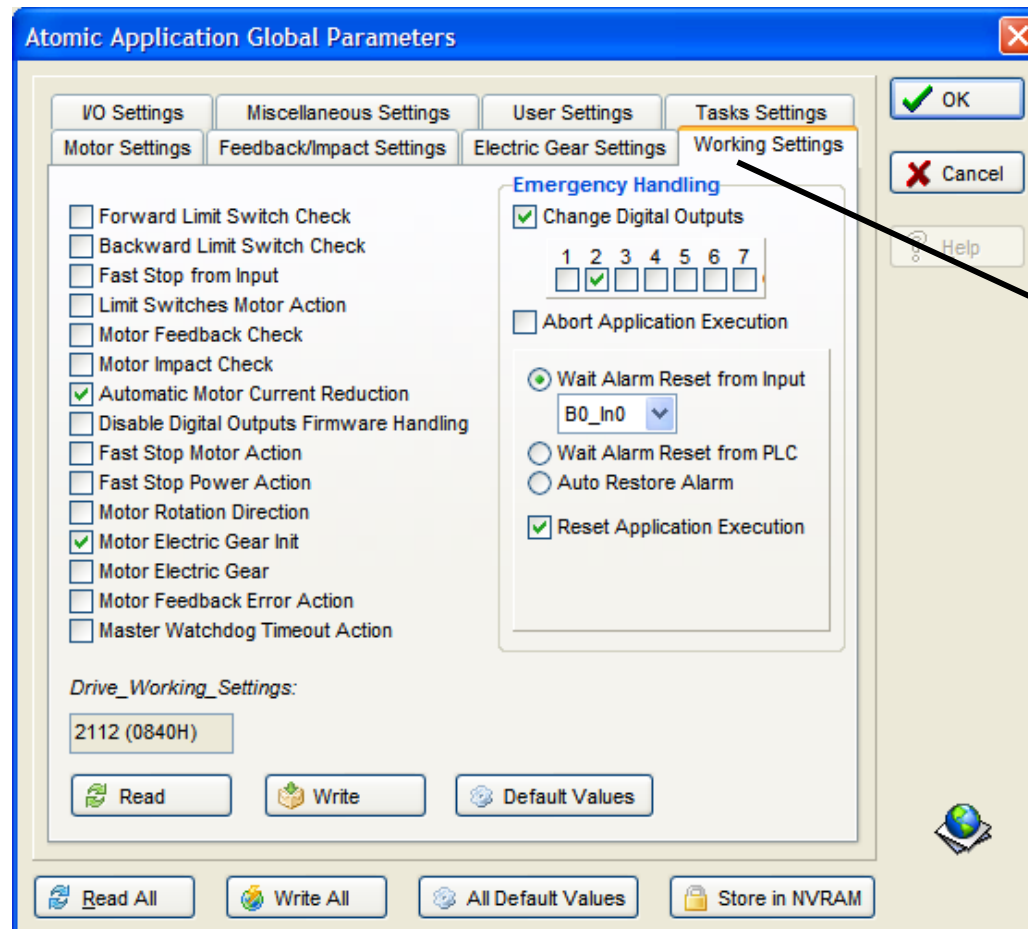
Attraverso la seguente finestra è possibile impostare i parametri iniziali dell'azionamento:



Tab per la visualizzazione dei parametri iniziali di settaggio dell'azionamento

Impostazioni di sistema

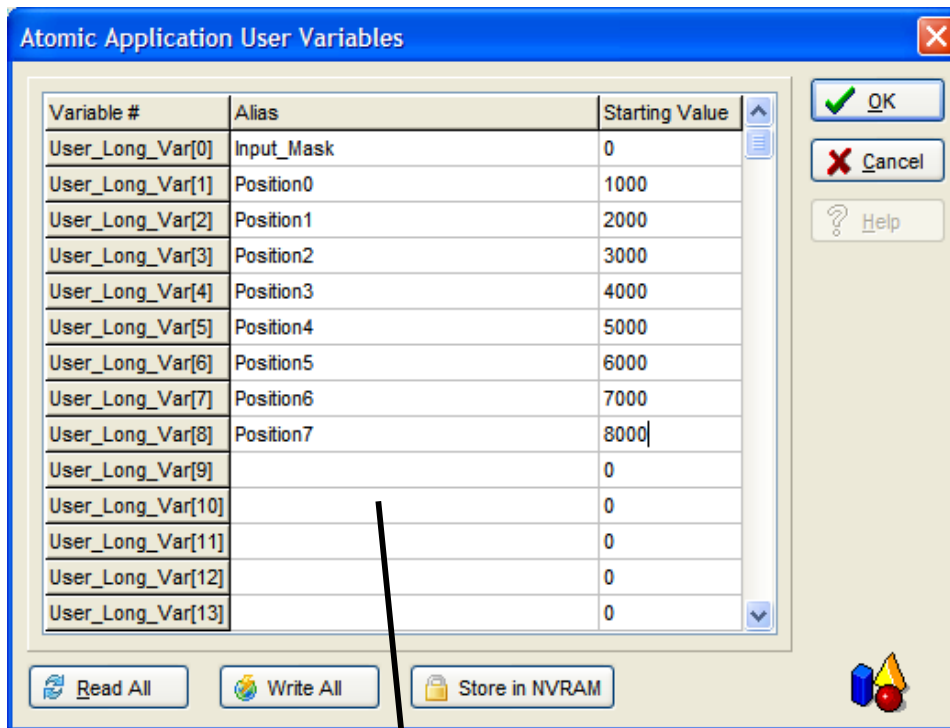
Attraverso la seguente finestra è possibile impostare i parametri iniziali dell'azionamento:



Tab per l'impostazione dei parametri iniziali dell'azionamento

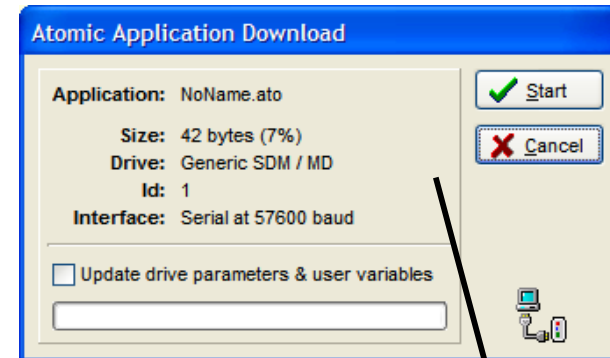
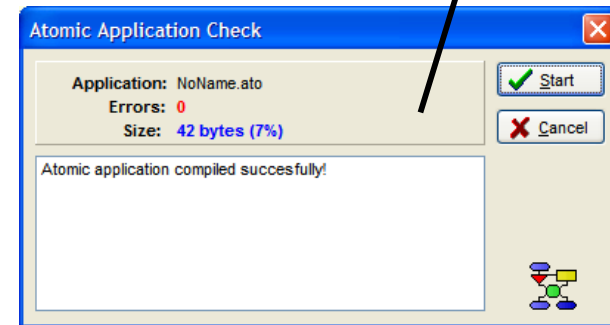
Impostazioni di sistema

Altre finestre di utilità:



Finestra per la gestione delle variabili utente

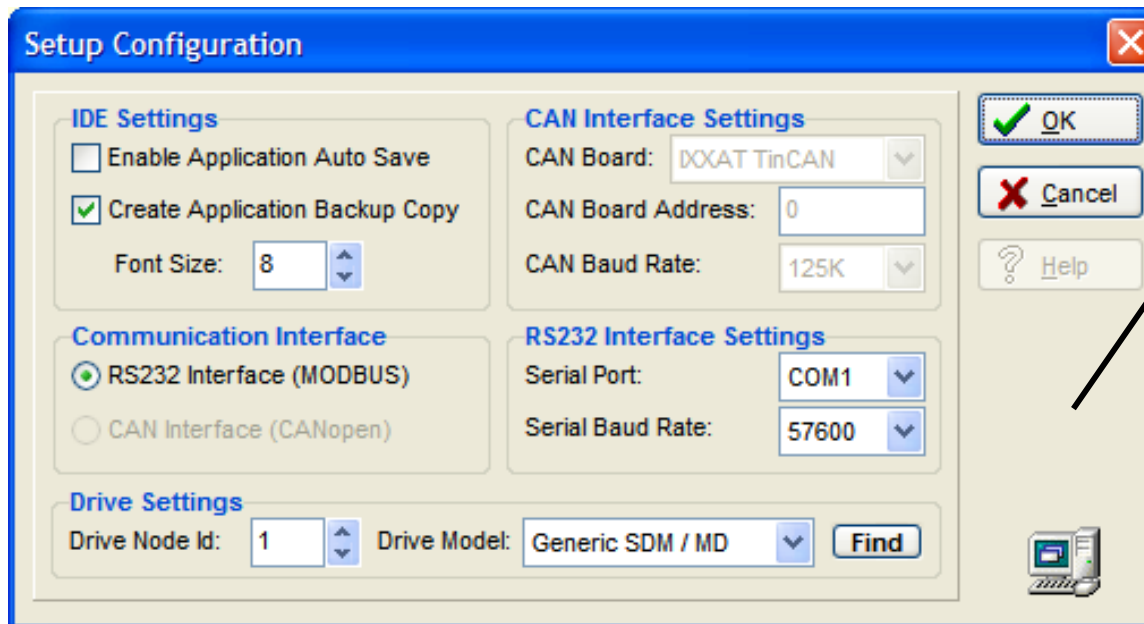
Finestra per il controllo della sintassi del programma



Finestra per il trasferimento dei programmi utente nell'azionamento

Configurazione ambiente PC

Altre finestre di utilità:



Finestra per settare i parametri del software Atomic