

```
int PDOProducerRequestProtocol(UNSIGN
{
    int          i;
    UNSIGNED16   j;
    CanMessage   outputMessage;

    outputMessage.id = LocalPDONumbe
    outputMessage.rtr = 0;
    outputMessage.length = PDODataAR

    for(j=0; j<outputMessage.length; j
        outputMessage.data[j] =

    i=CANwriteMessage(&Network, &out
```

## Caratteristiche Tecniche

### Software:

- Completa implementazione di tutte le funzionalità del CANopen Communication Profile DS301, servizi del CANopen Node (Slave)
- Suddivisione del codice nei 7 gruppi di servizi canonici del CANopen quali NMT, SYNC, TIME, EMCY, PDO, SDO e NodeGuarding
- Suddivisione del codice in 4 strati per l'implementazione di mini Kernel di sistema operativo, Protocol Layer, API Library e Applicazione utente
- Utility per la generazione automatica di codice C relativo ai files .eds (electronic data sheet) di configurazione del CANopen
- Facile integrazione dei Device Profiles DS40x
- Ampio set di programmi di esempio da utilizzare per l'integrazione o il porting su piattaforme proprietarie

### Piattaforme hardware:

- Tutte le schede ed i drivers di sistema operativo prodotti da TRAMA Srl sono compatibili con il codice CANopen Slave
- Software di supporto per i più diffusi CANcontroller

## Codice sorgente in linguaggio C del protocollo CANopen Slave Single Network

### Descrizione generale:

Il codice sorgente del protocollo CANopen Slave Single Network implementa fedelmente tutte le funzionalità previste dal documento di specifica edito dal consorzio del CAN In Automation (CiA) denominato DS301: CANopen Communication Profile.

In particolare, è fornito il supporto per tutti i 7 gruppi di servizi previsti dal protocollo quali NMT (Gestione della Rete), SYNC (Gestione delle sincronizzazioni), TIME (Gestione dei Real Time Clock), EMCY (Gestione degli eventi di emergenza), PDO (Gestione dei dati di processo), SDO (Gestione dei dati di servizio) e Node Guarding (Gestione della sicurezza sulla rete).

Per quanto riguarda la definizione dell'area di interfaccia tra protocollo e applicazione utente (Object Dictionary), il prodotto CW-SSN offre una esclusiva utility in grado di generare in modo automatico il codice in linguaggio C relativo al file di configurazione (electronic data sheet) previsto dalla specifica DS301.

### Portabilità:

Il pacchetto software è organizzato in blocchi funzionali indipendenti altamente scalabili, in modo da permettere la minimizzazione delle risorse hardware necessarie per l'implementazione dello stesso.

In particolare sono definibili, a livello di header files, tutti i parametri che hanno un impatto sulle strutture dati utilizzate dal protocollo.

Questa peculiare caratteristica permette di "confezionare" il protocollo CANopen sia per l'integrazione in sistemi con notevoli risorse hardware quali microprocessori dell'ultima generazione, sia in sistemi basati su mini microcontrollori embedded.

L'interfaccia verso l'hardware proprietario è assicurata da un insieme molto ristretto di chiamate a funzioni implementabili dall'utente.

## Architettura della libreria software

