



PROTOCOLLO DI COMUNICAZIONE DEGLI NSC

Specifica del protocollo di comunicazione del sistema Networked Shared Controller

Aug 11, 2016 - versione: 48

Introduzione

Il sistema di controllori programmabili a cellule distribuite retificate NSC, usa come protocollo di comunicazione di base lo SFBP (Simple Field Bus Protocol).

Lo SFBP costituisce uno stack che, con riferimento al modello ISO-OSI, non interferisce a livello applicativo, lasciandone libera la definizione.

Nel sistema NSC sono stati quindi utilizzati i 6 byte del Data Unit del pacchetto standard per definire i diversi tipi di informazione a livello applicativo, come stabilito dal modello ISO-OSI.

Pacchetto standard SFBP

header				Data Unit						tail
SM	destAddr	sndAddr	Packet info	D0	D1	D2	D3	D4	D5	CRC

I pacchetti SFBP si distinguono per il tipo definito dal byte Packet Information tra i quali figurano i pacchetti di controllo (CTRL_PACKET) e i pacchetti dati (DATA_PACKET).

Per maggiori informazioni sul protocollo SFBP consultare il documento “*SFBP simple field bus protocol v2.pdf*” disponibile all’indirizzo <http://www.smartcontroller.org>

Definizione generale dei dati presenti nel Data Unit a livello applicativo

Come precisato nell'introduzione, ottemperando alle disposizioni del protocollo SFBP che lascia a livello applicativo la definizione del contenuto e della relativa semantica, del Data Unit, nel sistema NSC sono stati organizzati secondo le seguenti linee generali:

- Si distinguono diversi tipi di messaggio, che vengono specificati dal byte D5 per i pacchetti di controllo (CTRL_PACKET);
- I primi 4 byte portano informazioni dipendenti dal contesto del messaggi, definiti dal byte D5;
- Il byte D4 può fornire informazioni aggiuntive, come un ID;
- I pacchetti Dati (DATA_PACKET) possono trasportare dati (usando tutto il Data Unit ed eventualmente più pacchetti concatenati) dipendenti dal contesto di una sessione aperta da un messaggio di controllo; tale pacchetto viene tipicamente originato dalla trasmittente del dato (o flusso di pacchetti dati) che ne fa seguito, o dalla ricevente del dato (o del flusso di dati) che si attende come risposta dalla trasmittente remota (vedi fig. 1)

Dati messaggio				Parametro opzionale	Tipo messaggio
D0	D1	D2	D3	D4	D5

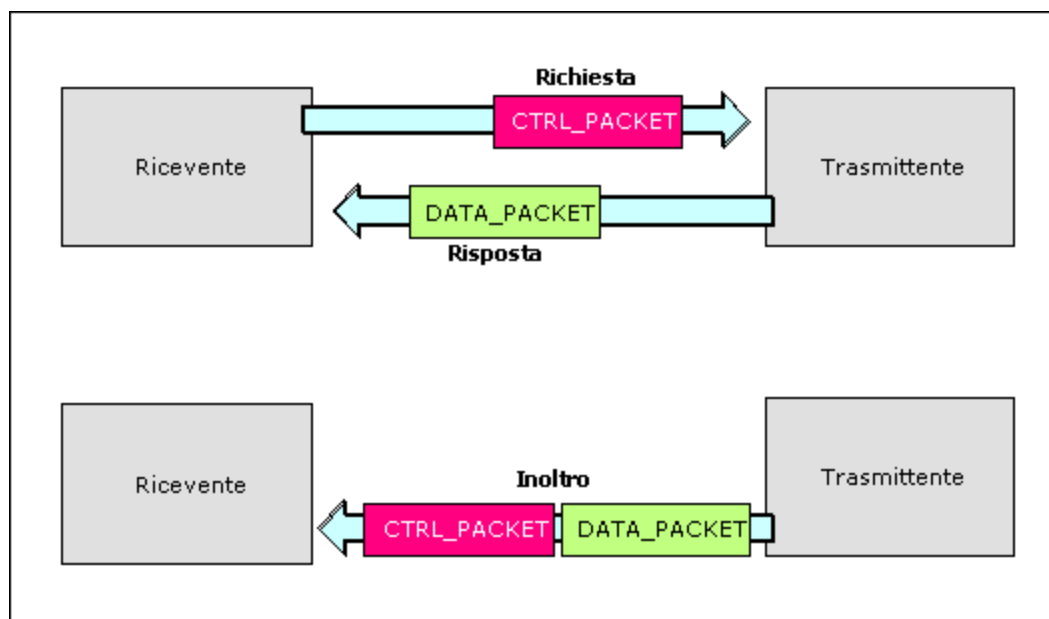


Figura 1

I pacchetti TIME_SET and TIME_GET trasportano in tutto il Data Unit le informazioni relative al tempo.

Messaggi

Nel sistema NSC i messaggi trasportano informazioni specifiche all'applicazione di sistema (ovvero messaggi di Sistema Operativo) e informazioni specifiche dell'applicazione utente (messaggi utente), ed essi vengono distinti dal byte D5 (tipo messaggio): Il valore dei tipi di messaggio (D5) di sistema sono compresi da 1 a 42, e da 226 a 255; il valore dei tipi di messaggio utente sono compresi tra 43 e 127, e tra 141 e 225. I valori tra 128 e 140 sono riservati.

Tipi di messaggio di sistema

(questi messaggi non vanno confusi con i pacchetti di sistema SFBP, che operano ad un livello inferiore nel modello ISO-OSI).

I messaggi sono identificati dal tipo (D5) e opzionalmente da un ID (D4) che dipende dal tipo messaggio.

Il sistema NSC prevede una disposizione dati per ingressi e uscite a blocchi di 16, ovvero 16 ingressi e 16 uscite per blocco, tali blocchi sono chiamati banchi (banks) e indirizzano sempre un insieme di I/O (max 16 in e 16 out), sono disponibili fino a 16 banchi permettendo di avere un massimo di 256 ingressi e 256 uscite indirizzabili per unità.

Nei messaggi di richiesta e ritorno dello stato di ingressi e uscite il campo ID (D4) indica il banco e altre informazioni di stato, occupando parte dei bit del byte come di seguito descritto:

<i>bank</i>							
<i>bit 7</i>	<i>6</i>	<i>5</i>	<i>4</i>	<i>3</i>	<i>2</i>	<i>1</i>	<i>bit 0</i>
bank				res.*	res.*	res.*	err

I bit 7:4 rappresentano il banco (0 a 15, 16 banchi), il banco di default è zero; i bit 3, 2 e 1 sono riservati e devono essere a zero; il bit 0 indica lo stato di errore/blocco o STOP dell'unità: 0 non in errore, 1 in errore. Non tutti i messaggi di stato ingressi/uscite ritornano questa informazione, in particolare i messaggi inviati all'unità non hanno questa informazione. In questi casi il bit deve essere a zero.

<i>Tipo messaggio</i>	<i>Valore byte D5</i>	<i>Descrizione</i>	<i>Struttura dati (solo sez. Data Unit)</i>			
			(lsb) Data 0	(lsb) Data 1	ID	Tipo
CTRLTYPE_SETTARGET	1	Imposta l'indirizzo a cui saranno inviate le notifiche sulle variazioni di ingresso (se specificato il relativo evento, per dispositivi soggetti ad evento) e le variazioni di uscita. Il D0 specifica l'indirizzo del target (l'unità che deve ricevere le notifiche).	target	-	-	1
CTRLTYPE_SETOUT	2	Imposta le uscite in base ai valori di Data 0 (usato come dato) e Data 1 (usato come mask). In ID:bank è fornito il banco di I/O interessato, default zero. ID:err (il bit errore) a zero.	data	mask	bank (vedi pag.3)	2
CTRLTYPE_DUMPERR	5	Richiesta di risposta dello stato del dispositivo, il pacchetto di risposta è di tipo DATA_PACKET (vedi riga seguente)	-	-	-	5

Tipo messaggio	Valore byte D5	Descrizione	Struttura dati (solo sez. Data Unit)					
			(lsb) Data 0		(lsb) Data 1		ID	Tipo
n.a.	n.a.	Risposta da una richiesta CTRLTYPE_DUMPERR. Il pacchetto è di tipo DATA_PACKET cpi = current program information pointer ip = instruction pointer ipBL:ipBH = absolute base address of instr. Pointer sp = (math) stack pointer err = error code						
			cpi	ip	ipBL	ipBH	sp	err
CTRLTYPE_GETIN	6	Richiesta di risposta con lo stato degli ingressi, il pacchetto di risposta è di tipo CTRL_PACKET. Il campo ID:bank specifica il banco da leggere (default zero), bit errore (ID:err) a zero.	-	-	-	-	bank (vedi pag.3)	6
CTRLTYPE_GETOUT	7	Richiesta di risponsta con lo stato delle uscite, il pacchetto di risposta è di tipo CTRL_PACKET. Il campo ID:bank specifica il banco da leggere (default zero), bit errore (ID:err) a zero.	-	-	-	-	bank (vedi pag.3)	7
CTRLTYPE_RPC_0PARAM	8...12	Richiesta esecuzione procedura remota (RPC). Il valore in D5 sottratto alla base 8 indica il numero di parametri passati con la chiamata e contenuti nel pacchetto: 8 = no parametri 9 = 1 parametro di tipo int 10 = 2 parametri di tipo int 11 = 3 parametri di cui il primo int e i successivi byte 12 = 4 parametri di tipo byte Il valore in D4 specifica il puntatore alla funzione.	par0	par1	par2	par3	*f ()	8..12
CTRLTYPE_PRGSTATE	244	Stato programmazione, usato SOLO per unità dotate di memorizzazione del programma utente in flash. Per dettagli vedere il protocollo di programmazione , di seguito in questo documento.	err	0	0	0	0	244
CTRLTYPE_RSTADDR	245	Reset address a NULL. L'azione di questo controllo azzera l'indirizzo dell'unità ponendola nello stato deindirizzata. da = Disabilita Autoindirizzamento, se 1 l'autoindirizzamento viene disabilitato (ove disponibile), oltre a deindirizzare l'unità.	da	-	-	-	-	245
CTRLTYPE_LNGCLICKSTATE	246	Inviato all'unità che si è registrata per la notifica (vedi CTRLTYPE_SETTARGET), in data0 viene riportato lo stato degli ingressi in cui è stato rilevato l'evento longclick in quell'istante in cui il pacchetto viene inviato.	clicked		0	0	0	246
CTRLTYPE_CLICKSTATE	247	Come per CTRLTYPE_LNGCLICKSTATE ma per evento click.	clicked		0	0	0	247
CTRLTYPE_INSTATE	248	Inviato all'unità che si è registrata per la notifica (vedi CTRLTYPE_SETTARGET) per unità con firmware inferiore a 1.19 (vedi anche CTRLTYPE_IOSTATE per unità con firmware >= 1.19) o in risposta al messaggio CTRLTYPE_GETIN. In data0 viene riportato lo stato degli ingressi, in data1 la mask degli ingressi che hanno avuto l'ultimo cambiamento. Il campo ID:bank riporta il banco di ingressi, il bit errore (ID:err) è impostato a 0 o a 1 se unità in blocco. <u>NOTA: Se l'unità è in blocco/errore o in STOP (ma non in STOP totale) questo messaggio viene inviato in broadcast ad ogni variazione degli ingressi, e il campo ID:err è impostato a 1.</u>	statoIn		maskIn		bank (vedi pag.3)	248

Tipo messaggio	Valore byte D5	Descrizione	Struttura dati (solo sez. Data Unit)					
			(lsb) Data 0		(lsb) Data 1		ID	Tipo
CTRLTYPE_OUTSTATE	249	Inviato in risposta al messaggio CTRLTTYPE_GETOUT oppure all'unità che si è registrata per la notifica (vedi CTRLTTYPE_SETTARGET) in seguito a variazioni sulle uscite logiche. Per unità con firmware >=1.19 invece viene inviato CTRLTTYPE_IOSTATE. In data0 viene riportato lo stato delle uscite, mentre in data1 la mask i cui bit alti indicano quali uscite sono variate (solo in caso di notifica). Se inviato in risposta a CTRLTTYPE_GETOUT il campo ID:bank fornisce il banco di uscite letto (default zero), il bit ID:err è posto sempre a zero con questo messaggio.	reso in risposta a CTRLTTYPE_GETOUT:					
			statoOut		0	0	bank <i>(vedi pag.3)</i>	249
			reso su variazione delle uscite logiche:					
			statoOut		maskOut			
			LSB	MSB	LSB	MSB		249
CTRLTYPE_GETSERIAL	250	Richiesta del numero seriale, tipo di unità e versione del firmware. La risposta è un pacchetto di tipo DATA_PACKET con i dati richiesti (vedi riga che segue)						
			-	-	-	-	-	250
n.a.	n.a.	Inviato in risposta alla ricezione di un messaggio CTRLTTYPE_GETSERIAL; il messaggio è di tipo DATA_PACKET. Serial 0,1,2 = byte codice seriale; Dev.ID=id del tipo di unità; Vers. Major, minor = versione firmware.						
			Serial 0	Serial 1	Serial 2	Dev. ID	Vers. Major	Vers. minor
CTRLTYPE_SETADDR	251	Imposta l'indirizzo logico dell'unità. Serial 0,1,2 deve avere il codice seriale corrispondente all'unità, ADR deve contenere l'indirizzo logico. Se il codice seriale non corrisponde il messaggio viene ignorato, diversamente viene assegnato l'indirizzo logico e l'unità si pone in stato di STOP.						
			Serial 0	Serial 1	Serial 2	Adr	-	251
CTRLTYPE_CLEARERR	252	Azzera lo stato di errore, se presente. Lo stato di STOP è analogo ad uno stato di errore.						
			-	-	-	-	-	252
CTRLTYPE_REPROGRAM	253	Predispone l'unità per ricevere il programma utente. Se l'unità è già in stato di programmazione, il messaggio viene ignorato. In data0 viene riportata la dimensione in bytes del programma, chk riporta la checksum dei CRC di ciascuno dei pacchetti che saranno inviati. Il programma viene inviato immediatamente dopo mediante un flusso di pacchetti di tipo DATA_PACKET. Per dettagli vedere il protocollo di programmazione , di seguito in questo documento.						
			Dimens. programma					
			LSB	MSB	chk	-	-	253
CTRLTYPE_REBOOT	254	Riavvia il sistema operativo. Se il D0 del pacchetto contiene il valore 55 o 63 l'unità viene anche posta in stato di STOP. In stato di STOP l'unità invia in broadcast ogni variazione di stato di ingressi e uscite. Se però viene inviato una seconda volta questo messaggio con in D0 il valore 55, l'unità viene posta in tacitazione e nessun messaggio sarà trasmesso se variano gli ingressi o le uscite. Questa condizione è temporanea e si annulla al primo riavvio, sia inviando questo messaggio con D0 a zero, sia per spegnimento e riaccensione dell'unità.						
			Stop?	-	-	-	-	254

Tipo messaggio	Valore byte D5	Descrizione	Struttura dati (solo sez. Data Unit)					
			(lsb) Data 0		(lsb) Data 1		ID	Tipo
CTRLTYPE_STORE	243	Predisporre l'unità per ricevere un flusso di dati che saranno memorizzati nella eeprom nell'area dati utente, tipicamente per registrare una tabella di tempi dello schedatore. Dopo questo messaggio, l'unità attende di ricevere i dati fino al raggiungimento specificato nei primi due byte. Ad ogni pacchetto ricevuto l'unità risponderà con un messaggio CTRLTYPE_OK. Il byte di controllo A permette di specificare l'area da scrivere. Il byte inferiore deve essere zero, salvo dove specificato. A = 0 memorizza dati come schedatore. A = 1 memorizza 6 byte (1 pacchetto dati) di parametri DST (daylight saving time) A = 2 memorizza la configurazione I/O usando il byte inferiore (cfgIO, byte 2) del pacchetto di controllo, non riceve alcun altro pacchetto dati successivo. A = 3...127 riservati; A = 128 memorizza dati nell'area file utente in eeprom, il byte 2 e 4 permettono di impostare un offset rispetto l'indirizzo 0 dell'area file. NOTE: L'opzione A=1 è disponibile solo su dispositivi dotati di orologio, altre opzioni a seconda del modello. Opzione A=128 disponibile solo da vers. 1.20. Se si tenta di inviare un comando con parametri non validi il comando viene rifiutato ma nessun messaggio di errore viene reso: è compito dell'applicazione fornire valori legali.	Dimens. dati		controllo			
			LSB	MSB	0	A	-	243
			LSB	MSB	cfgIO	2	-	243
			LSB	MSB	offset L	128	offset H	243
CTRLTYPE_OK	242	Inviato in risposta ad una operazione sequenziata, come quella determinata dal messaggio CTRLTYPE_STORE, riportando in <i>offset</i> la posizione correntemente raggiunta ed in <i>errore</i> eventuali stati di errore (non fatali) di programma. Inviato in risposta ad un messaggio RPCSYNC e successivo messaggio RPC dopo che l'esecuzione della chiamata a funzione è stata completata. Il ritorno della funzione viene reso nel campo <i>errore</i>.	errore		offset			
			LSB	-	LSB	MSB	0	242
CTRLTYPE_GETSTORE	241	Ritorna al mittente 5 byte di dati relativi alla memoria eeprom utente, a partire dalla posizione indicata in POS, il byte 5 (ultimo byte) del pacchetto reso indica il numero di byte effettivamente resi. Il pacchetto reso è di tipo DATA_PACKET. Dalla versione firmware 1.20 il byte D2 (LOC) permette di specificare una locazione di base: LOC = 0 (default) imposta offset dal byte zero; LOC = 1 imposta offset lettura area schedatore. LOC = 2 locazione POS assoluta (dalla versione firmware 1.23) LOC = (altri valori riservati per future estensioni) NOTA: E' possibile leggere qualsiasi posizione della eeprom, incluso sezioni esterne al file utente e alla schedulazione (qualora prevista), con la sola eccezione delle aree di sistema che non sono accessibili.	POS					
			LSB	MSB	-	-	-	241
			Dalla versione firmware 1.20:					
			POS		D2	D3	D4	D5
			LSB	MSB	LOC	0	-	241
n.a.	n.a.	Pacchetto dati (tipo DATA_PACKET) inviato in risposta di un messaggio CTRLTYPE_GETSTORE. I primi 5 byte contengono dati, l'ultimo byte (cb) indica il numero effettivo di byte validi contenuti nel messaggio.	data0	data1	data2	data3	data4	cb

Tipo messaggio	Valore byte D5	Descrizione	Struttura dati (solo sez. Data Unit)			
			(lsb) Data 0	(lsb) Data 1	ID	Tipo
CTRLTYPE_SETSERIAL	240	Imposta il codice seriale del dispositivo ovvero l'indirizzo MAC. I primi 3 byte forniscono il codice seriale. NOTA: <i>Il codice seriale viene accettato solo se:</i> <i>- l'unità è vergine (mai programmata prima, da fabbrica); oppure</i> <i>- l'unità ha un indirizzo logico (nonzero) e il comando è specificatamente inviato all'indirizzo dell'unità.</i>	Serial			
			LSB	MSB	-	240
CTRLTYPE_OUTVALUE	246	Inviato all'unità che si è registrata per la notifica (vedi CTRLTYPE_SETTARGET) su variazioni del valore di uscite analogiche o dimmer. In data0 viene riportato il valore dell'uscita e in data1 il canale interessato. NOTA: <i>Questo pacchetto non viene inviato dal sistema operativo, ma è gestito a livello applicativo del programma utente.</i>	outvalue		channel	
			LSB	MSB	LSB	246
CTRLTYPE_IOSTATE	234	Disponibile solo dalla versione 1.19 del firmware, notifica lo stato di ingressi e uscite in luogo dei messaggi CTRLTYPE_INSTATE e OUTSTATE. Viene segnalato automaticamente al target specificato con CTRLTYPE_SETTARGET e solo se il target è specificato (non zero), ogni volta che avviene una variazione di ingresso o uscita, sia endogena che esogena. L'invio non è istantaneo salvo non sia specificato il flag IOS_IMM_NTFY nella configurazione ingressi (e solo nei modelli dove è disponibile), l'invio avviene dopo circa 250 millisecondi di stato stabile di ingressi e uscite. In Data0 sono riportati gli ingressi, in Data1 le uscite e il campo ID:bank il banco di I/O interessato, mentre il bit errore ID:err è a 1 se l'unità è in blocco/errore o in STOP (ma non in STOP totale/silenzioso). Se avviene una variazione su più banchi saranno inviati più messaggi, ciascuno per il proprio banco, anche se la variazione coinvolge solo un ingresso o solo una uscita del banco.	inputs		outputs	
			LSB	MSB	LSB	234
CTRLTYPE_RPCSYNC	235	Inviato all'unità verso la quale segue immediatamente una chiamata RPC (vedi CTRLTYPE_RPC0PARAM), richiede conferma ad esecuzione avvenuta mediante messaggio CTRLTYPE_OK	0	0	0	235
CTRLTYPE_FIRMWAREUPLOAD	236	Imposta il dispositivo in system mode permettendo l'eventuale invio del firmware. L'uscita richiede un reboot system (vedi messaggi system) o la rimozione dell'alimentazione. Se il messaggio viene preceduto da uno CTRLTYPE_RSTADDR, si avvia l'autoindirizzamento.	0	0	0	236
CTRLTYPE_POWERRESTORE	237	Segnala che è possibile uscire dal modo powersave, imposta il dispositivo che lo riceve a uscire dal modo powersave. In base al modello i relé possono ripristinarsi allo stato corrente previsto.	0	0	0	237
CTRLTYPE_POWERSAVE	238	Segnala che è necessario entrare in modo powersave (riduzione consumo energia, in base al modello potrebbe rilasciare i relé se l'unità ne è dotata, lo stato non viene perduto però).	0	0	0	238
CTRLTYPE_RCLICK	239	Comanda l'evento click per l'ingresso specificato in LSB Data 0 (input 0, 1, 2, ...ecc.). L'evento viene quindi eseguito come se fosse stato rilevato sul corrispondente ingresso. Esempio: 3,0,0,0,0,239 comanda evento click per ingresso 3.	Input click	0	0	239

<i>Tipo messaggio</i>	<i>Valore byte D5</i>	<i>Descrizione</i>	<i>Struttura dati (solo sez. Data Unit)</i>			
			(lsb) Data 0	(lsb) Data 1	ID	Tipo

Tipi di messaggio utente

I messaggi utente sono specifici dell'applicazione utente e sono liberi, in essi solo il tipo (byte D5) viene specificato, entro le gamme riservate a tale scopo, ed il loro significato definito dalla stessa applicazione utente.

Protocollo di memorizzazione dati in eeprom

La memorizzazione in eeprom di dati è utilizzata dalle unità dotate di cronogestore per la memorizzazione della tabella di schedulazione, ma in futuro potrebbe essere utilizzata per altre funzionalità che richiedano un file dati utente.

L'invio dei dati da parte di un pari remoto avviene con l'invio di un pacchetto di controllo (CTRL_PACKET) con il messaggio CTRLTYPE_STORE con la dimensione, in byte, dei dati da inviare; a questo deve immediatamente fare seguito una serie di pacchetti dati (DATA_PACKET) con i dati: saranno inviati tanti pacchetti quanti ne sono necessari al fine di trasferire il file accomodando di volta in volta il numero di byte massimo trasmissibile o, in caso di numero rimanente minore, inviando il numero di byte validi. Ad ogni pacchetto dati inviato la trasmittente deve attendere una risposta dalla ricevente mediante un pacchetto di controllo con messaggio CTRLTYPE_OK.

Un meccanismo di timeout da entrambe le parti (o altro mezzo) impedisce che si possa verificare una situazione di stallo in caso di mancata risposta o mancato invio di dati.

Il trasferimento si completa quando sia stato trasmesso, e la ricevente ne abbia confermato la ricezione, dell'ultimo pacchetto dati tale da raggiungere la dimensione in byte prevista per la trasmissione.

Protocollo di programmazione

Le unità NSC possono essere dotate di un'area di memoria eeprom o flash per immagazzinare il programma utente. In base al tipo di memoria impiegata, identificabile dalla versione del firmware, si hanno due diverse modalità di trasmissione del programma.

La memorizzazione del programma utente in flash, sia per le caratteristiche fisiche del supporto, sia per l'estensione che tale programma può assumere, richiede il rispetto di tempi non determinabili esternamente al microprocessore coinvolto nelle operazioni di scrittura della memoria flash. Per queste ragioni il protocollo per la memorizzazione in flash differisce da quello usato per la memorizzazione in eeprom, come di seguito descritto.

Si noti che entrambe le modalità di programmazione fanno ricorso allo stesso messaggio che avvia la procedura di programmazione.

Protocollo per la memorizzazione in eeprom

Il processo viene avviato con la ricezione di un pacchetto di tipo CTRL_PACKET, mediante il messaggio tipo CTRLTYPE_REPROGRAM. Immediatamente dopo devono fare seguito uno o più pacchetti di tipo DATA_PACKET mediante il programma da memorizzare (frammentato in vari pacchetti), a partire dalla locazione di memoria zero, relativa all'area programma utente. I pacchetti vengono trasmessi in modalità connessa (vedi specifiche SFBP). Ad ogni pacchetto ricevuto viene prorogato un timeout, scaduto il quale e senza che la ricezione sia stata completata con successo, si interrompe da parte della ricevente lo stato di Riprogrammazione, l'intera operazione viene abortita e si pone in stato di Errore di riprogrammazione.

Nel caso normale che tutti i pacchetti siano stati ricevuti e la checksum dei CRC di ogni pacchetto sia risultata corretta, viene rimosso lo stato di Riprogrammazione e l'unità viene posta in stato di STOP.

Lo stato di STOP prevede l'invio automatico di messaggi CTRLTYPE_OUTSTATE su ogni variazione di uscita (ad esempio in seguito a comando mediante pacchetto CTRLTYPE_SETOUT), e CTRLTYPE_INSTATE su ogni variazione di ingresso; oppure CTRLTYPE_IOSTATE sui modelli più recenti con la sintesi degli ingressi e uscite in caso di variazione di uno di essi.

Protocollo per la memorizzazione in flash

Il processo viene avviato con la ricezione di un pacchetto di tipo CTRL_PACKET, mediante il messaggio tipo CTRLTYPE_REPROGRAM. Immediatamente dopo devono fare seguito uno o più pacchetti di tipo DATA_PACKET mediante il programma da memorizzare, a partire dalla locazione di memoria zero, relativa all'area programma utente (quindi una posizione spostata di un offset fisso specificato all'interno del firmware dell'unità). I pacchetti vengono trasmessi in modalità connessa (vedi specifiche SFBP).

L'unità trasmittente deve inviare il programma frammentandolo in parti di massimo 6 bytes per coprire la capacità di ciascun singolo pacchetto, ed una volta trasmesso ogni pacchetto deve attendere il messaggio CTRLTYPE_PRGSTATE prima di continuare.

Nella ricevente, ad ogni pacchetto ricevuto viene prorogato un timeout, scaduto il quale e senza che la ricezione proceda, si interrompe da parte della ricevente lo stato di Riprogrammazione, l'intera operazione viene abortita e si pone in stato di Errore di riprogrammazione, nessun messaggio di risposta viene inviato alla trasmittente.

Nel caso normale di corretta ricezione, la ricevente raccoglie a blocchi i dati ricevuti in modo da riempire un buffer temporaneo di capienza pari ad una pagina di memoria flash (tipicamente 128 bytes). Una volta riempito questo buffer la ricevente provvede a trasferire i dati del buffer nella memoria flash, tenendo conto dell'ultima pagina memorizzata; quindi invia il messaggio CTRLTYPE_PRGSTATE con in D0 l'eventuale errore di memorizzazione.

In ogni caso se il buffer non è ancora pieno, la ricevente trasmette ad ogni pacchetto ricevuto un messaggio CTRLTYPE_PRGSTATE con in D0 l'ultimo codice di errore.

I possibili codici di errore in questa fase sono:

<i>Errore</i>	<i>Descrizione</i>
errNONE	Nessun errore, inviare prossimo dato
errFLASHFAIL	Errore memorizzazione in flash: indirizzo memoria non valido
errFLASHFAILURE	La memoria flash è danneggiata: il valore scritto non corrisponde con il valore letto
errSTOP (errREADADDRESS)	Programmazione completata, unità in STOP in attesa di riavvio

Nel caso normale che tutti i bytes siano stati ricevuti e la checksum dei CRC di ogni pacchetto sia risultata corretta, viene rimosso lo stato di Riprogrammazione e l'unità viene posta in stato di STOP, viceversa l'unità fa scadere il timeout, l'intera operazione viene abortita e si pone in stato di Errore di riprogrammazione, nessun messaggio di risposta viene inviato alla trasmittente.

Anche se al completamento con successo della ricezione del programma l'unità dovrebbe rispondere con un messaggio CTRLTYPE_PRGSTATE dove D0 ha valore errREADADDRESS, la trasmittente terminato l'invio del programma, dovrebbe comunque usare un messaggio CTRLTYPE_DUMPERR per verificare che l'unità sia in errREADADDRESS, e non abbia altri errori.