

MOBA Mobile Automation AG

Spezifikation

ADuC836_SPISendPtr()

aus der ADuC836-Bibliothek

Version 1.016

Produkt	MRW 4-20mA	
	(Momenten unabhängige Redundante Wägezelle)	
Auftraggeber	MOBA Mobile Automation AG Kapellenstraße 15 65555 Limburg Germany	
Auftragnehmer	MOBA Mobile Automation AG Kapellenstraße 15 65555 Limburg Germany	

Dokument erstellt von	Datum	Unterschrift
M.Offenbach	12.05.2022	

MRW 4-20mA vertraulich

Diese Dokumentation des Unittests basiert auf einem Vordruck der MOBA AG.

Der Inhalt darf ausschließlich den am Projekt beteiligten Personen zugängig gemacht werden. Insbesondere die Weitergabe an Dritte ist ohne ausdrückliche schriftliche Erlaubnis der MOBA AG nicht erlaubt.

Außerhalb des gemeinsamen Projektes darf kein Teil dieser Unterlagen für irgendwelche Zwecke vervielfältigt oder übertragen werden, unabhängig davon, auf welche Art und Weise oder mit welchen Mitteln dies geschieht.

Die hier getroffenen Festlegungen schließen nicht aus, dass in einer gesonderten Geheimhaltungsvereinbarung weiterreichende oder abweichende Vereinbarungen zur Wahrung der Vertraulichkeit getroffen und festgeschrieben werden.

Copyright by

MOBA Mobile Automation AG Kapellenstr. 15 D-65555 Limburg Internet: www.moba.de





Inhaltsverzeichnis

1	Einf	Einführung4			
	1.1	Vorwort	4		
	1.2	Änderungshistorie	4		
	1.3	Ansprechpartner	5		
	1.4	Anhänge	5		
		Glossar			
2	ADı	uC836_SPISendPtr()	6		
	2.1	Beschreibung	6		
		Spezifikation			
		mmentare			
4	Anh	nang	Ç		



1 Einführung

1.1 Vorwort

Die MOBA AG versteht sich als Partner für die Entwicklung und Lieferung kundenspezifischer Elektronikkomponenten und daraus zusammengestellter Steuerungssysteme, die für den Einsatz an mobilen Maschinen konzipiert sind.

Die hier vorliegende Spezifikation beschreibt das exakte Verhalten der Bibliotheksfunktion ADuC836_SPISendPtr() der Datei ADuC836_SPISendPtr.c

Dies beginnt mit der Angabe der Übergabeparameter sowie dem Rückgabewert der Funktion. Es folgen dann die Beschreibungen des Verhaltens der Funktion

Jede Beschreibung wird indiziert festgehalten. Somit ist in weiteren Dokumenten leicht Bezug auf die Spezifikation zu nehmen.

1.2 Änderungshistorie

Version	Datum	Kapitel	Änderung / Ergänzung
1.0	12.05.2022	alle	Erstellung

Seite 4 von 9 Spezifikation Version 1.0

vertraulich MRW 4-20mA



1.3 Ansprechpartner

MOBA Mobile Automation AG

Kapellenstraße 15 65555 Limburg

Name	Position	Telefonnummer	E-Mail
Boris Zils	Produktmanager	+49(0)6431-9577- 123	b.zils@moba.de
Sebastian Schlesies	Vertrieb	+49(0)6431-9577- 267	s.schlesies@moba.de
Jürgen Stiller	Entwicklungsleiter	+49(0)6431-9577- 282	j.stiller@moba.de
Norbert Lipowski	Entwicklung	+49(0)6431-9577- 137	n.lipowski@moba.de

1.4 Anhänge

Dokumentname	Beschreibung

1.5 Glossar

Abkürzung / Fachbegriff	Beschreibung / Definition	
MRW	Momenten unabhängige Redundante Wägezelle	
DMS	Dehnungsmessstreifen	



2 ADuC836_SPISendPtr()

2.1 Beschreibung

ADuC836_SPISendPtr() sendet zeichenweise die Zeichen aus einem Speicherbereich aus. Hierzu bedient man sich der Funktion , ADuC836_SPISendChar()' zum Senden eines einzelnen Zeichens.

Aufgrund von Ergänzungsmaßnahmen in der SPI-Interruptroutine ergibt sich für diese eine längere Laufzeit. Dies wiederum hat zur Folge, dass bereits vor Abarbeitung des Interrupts, das nächste Zeichen gesendet wird. Dies kann demzufolge nicht empfangen werden.

Um das Problem zu beheben, ist nun in der zentralen Sendefunktion zur SPI "ADuC836_SPISendPtr()" nach dem Versand des Zeichens eine Pause von etwa 200us einzulegen.

Hierzu ist die Definition ,ADUC836_SPI_GLOBALS_SEND_WITH_BYTEPAUSE' in der Datei SPI_Global.h zu treffen und in der gleichen Datei den Wert der Definition , ADUC836_SPI_GLOBALS_BYTEPAUSE' auf 20 zu setzen.

vertraulich MRW 4-20mA



2.2 Spezifikation

Alle Spezifikationen sind in aufsteigender Reihenfolge zu erfüllen!

	ADuC836_SPISendPtr()			
Index	Parameter	Datentyp		
22.2.0.0	Zeiger auf den Sendepuffer	unsigned char*		
	Anzahl der zu sendenden Zeichen	unsigned int		
	(im Falle eines zu sendenden Strings darf dieser	and grown		
	Wert auch 0 sein)			
	Rückgabe	Datentyp		
22.2.1.0	Sende-Status	char		
	0: Puffer erfolgreich gesendet			
	1: Fehler ,Timeout'			
	Verhalten	Bemerkung		
22.2.2.0	Der Rückgabewert 'byRetVal' ist zu Beginn der	Initialisierung		
	Funktionsausführung mit 0 zu initialisieren.			
22.2.2.1	Ist die Anzahl der zu sendenden Zeichen = 0, ist	Anzahl der zu sendenden Zeichen ermitteln.		
	diese aus der Länge der Zeichenkette im Sendepuffer zu ermitteln.			
22.2.2.2	Es sind Vorbereitungen für den anschließenden	Vorbereitungen für den anschließenden		
	Empfang zu treffen. Dazu erfolgt der Aufruf der	Empfang treffen		
	Bibliotheksfunktion	. 0		
22.2.2.3	,ADuC836_SPIReleaseReception()' SPI-Mastermodus einschalten. Dies geschieht	Bürde abgeklemmt – Abweichungszähler setzen		
22.2.2.3	durch Aufruf der Funktion ,	Buide abgekienini – Abweichdrigszanier setzen		
	ADuC836_SPISwitch2Mode(ADUC836_SPI_MAS	ADUC836_SPI_MASTER_MODE = 0x10		
	TER MODE)	ADOCOSO_SFI_IVIASTEN_IVIODE = 0x10		
22.2.2.4	Nacheinander sind nun die Zeichen aus dem	Sendepuffer ausgeben		
	Sendepuffer zu übertragen. Hierzu bediene man			
	sich der Funktion ,ADuC836_SPISendChar()' und			
	legt deren Rückgabe in der Variablen 'byRetVal'			
	ab.			
22.2.2.5	Nach dem Senden eines Zeichens ist eine Pause	Sendepause nach dem Senden eines Zeichens		
	von etwa 200us einzulegen. Dies geschieht über	einlegen		
	das ,ADUC836_SPI_GLOBALS_BYTEPAUSE'-	ADUC836_SPI_GLOBALS_WITH_BYTEPAUSE		
	fache Inkrementieren einer zuvor mit 0	ist definiert		
	initialisierten Laufvariablen	ADUC836_SPI_GLOBALS_BYTEPAUSE = 20		
22.2.2.6	Wurde beim Senden des Zeichens ein Fehler	Sendefehler erkannt		
	entdeckt, ist die Sendeschleife nach der			
	Sendepause zu verlassen.			
22.2.2.7	SPI-Slavemodus einschalten. Dies geschieht durch	Sendevorgang abschließen		
	Aufruf der Funktion			
	,ADuC836_SPISwitch2Mode(ADUC836_SPI_SLA	ADUC836_SPI_SLAVE_MODE = 0x00		
	VE_MODE)'			
22.2.2.8	Rückgabe der Variablen 'byRetVal'.	Rückgabe		



3 Kommentare

vertraulich MRW 4-20mA



4 Anhang