



MSP430 - Intelligenter Stromsparer

Teil 1

Die MSP430-Mikrocontroller-Familie von Texas Instruments ist aufgrund des geringen Strombedarfs bestens für die Realisierung batteriebetriebener Geräte geeignet. Die hiermit startende Artikelserie stellt die MSP430-Familie vor und zeigt mittels Beispielen in den folgenden Teilen, wie der Mikrocontroller eingesetzt werden kann.

Strom sparen

Kennen Sie den großen Nachteil von batteriebetriebenen Geräten? - Genau, irgendwann ist die Batterie entladen. Sie stimmen sicherlich zu, dass dies bei Ihrer Armbanduhr, der TV-Fernbedienung oder einem Wecker nicht so oft passieren sollte. Wesentlich kritischer sind jedoch batteriebetriebene Zähler, wie beispielsweise Heizkostenzähler oder Wasserzähler. Diese Geräte müssen mehrere Jahre mit der fest eingebauten Batterie funktionieren. Texas Instruments hat deshalb Anfang der 90er-Jahre begonnen, eine Mikrocontrollerfa-

milie zu entwickeln, deren geringer Stromverbrauch den Aufbau solcher Zähler ermöglicht. Die MSP430-Mikrocontrollerfamilie erlaubt es zum Beispiel, Heizkostenzähler über einen Zeitraum von 10 Jahren aus einer Batterie zu versorgen.

Damit kennen wir auch schon die hervorsteckende Eigenschaft der MSP430-Mikrocontroller – den geringen Stromverbrauch. Die Stromaufnahme des Typs MSP430F1121 beträgt beispielsweise nur 160 μA bei einer Versorgungsspannung von 2,2 V und einer Taktfrequenz von 1 MHz. Dabei sind die CPU sowie einige periphere Module des Chips aktiv. Module, deren Funktionen nicht ständig verfüg-

bar sein müssen, sind abschaltbar. Dadurch kann man den Stromverbrauch des Chips minimieren. Beispiele für abschaltbare Peripheriemodule sind der analoge Komparator (Comparator_A) oder auch das A/D-Wandler-Modul ADC12. Angaben über den zusätzlichen Stromverbrauch dieser Module kann man dem chipspezifischen Datenblatt entnehmen.

Da der Mikrocontroller bei den meisten Anwendungen nicht ständig Aufgaben bearbeiten muss, wurden in der MSP430-Familie auch Low-Power-Modes implementiert. Es existieren bis zu 5 verschiedene Low-Power-Modes, bei denen die CPU oder einzelne chipinterne Taktsignale abgeschaltet werden. Dadurch reduziert sich der Stromverbrauch des Mikrocontrollers beträchtlich (bis unter 1 μA !).

Einen kleinen Beitrag zur Reduzierung des Stromverbrauches liefert auch die schnelle „Einschaltzeit“ der MSP430-Mikrocontroller. In maximal 6 μs wird der Chip vom Low-Power-Mode in den Aktiv-Mode geschaltet. Damit ist das mit dem Chip betriebene System sehr schnell voll einsatzbereit.

Das „Strom sparen“ der MSP430-Mikrocontroller setzt sich also aus den Komponenten geringer Aktivstrom, abschaltbare Peripheriemodule, Low-Power-Modus und einer schnellen Einschaltzeit zusammen.

Verschiedene MSP430-Typen

Die oben beschriebenen Eigenschaften sind für alle MSP430-Typen gültig. Die Unterschiede der verschiedenen Typen sind in Abbildung 1 dargestellt. Momentan können drei Gruppen von MSP430-Mikrocontrollern unterschieden werden. Bei den Derivaten MSP430P31x, MSP430P32x und MSP430P33x (diese Gruppe wird „MSP430P3xx“ genannt) handelt es sich um OTPs (= One Time Programmable), die auch mit einem LCD-Treiber ausgestattet sind. Die Typen MSP430F11x1, MSP430F12x, MSP430F13x und MSP430F14x (die Gruppe wird kurz „MSP430F1xx“ genannt) sind Mikrocontroller mit einem Flash-Speicher. Charakteristisch ist zudem, dass die MSP430F1xx-Controller keinen LCD-Treiber besitzen. Das aktuellste Derivat MSP430F41x wird jedoch mit Flash-Speicher und integriertem LCD-Treiber angeboten. Die typischen peripheren Module, die in die verschiedenen Chips integriert sind, kann man Abbildung 1 oder auch den einzelnen Datenblättern [2] entnehmen.

Bei den oben genannten Chipnamen taucht immer wieder der Buchstabe „x“ auf. Tatsächlich findet man in der Controllerbezeichnung anstelle des Buchstabens eine Ziffer zwischen 0 und 9. Diese Zahl ist

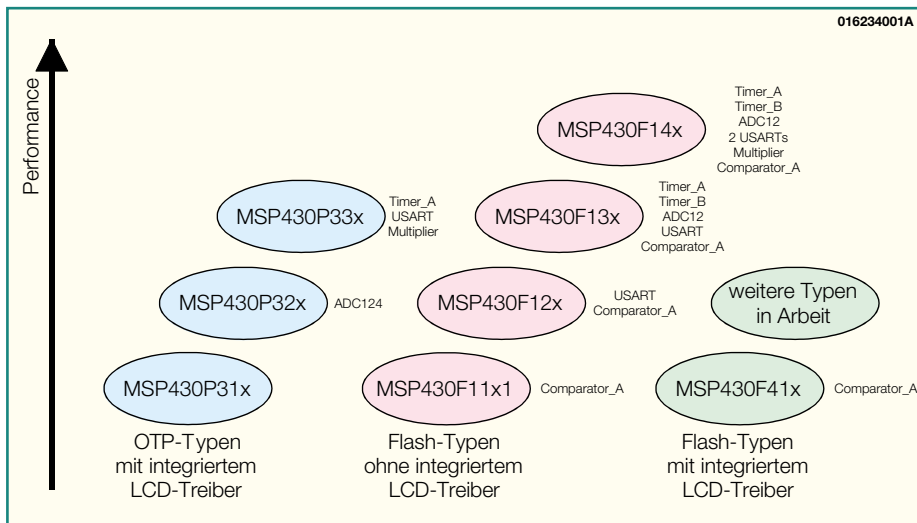


Bild 1: Die MSP430-Mikrocontrollerfamilie im Überblick

eine Codierung für die Größe des Programmspeichers. Tabelle 2 zeigt, welche Ziffer für welche Speichergröße verwendet wird.

Alle verfügbaren Varianten, insbesondere die Speichergröße betreffend, sind in der „Product List“ im Internet [1] sowie den Datenblättern [2] zu finden.

In diesem und den folgenden Artikeln werden wir insbesondere auf den neuen Typ MSP430F14x, dessen Blockschaltbild in Abbildung 3 dargestellt ist, eingehen.

Herzschlag des Mikrocontrollers

Neben der CPU selbst ist wohl das Modul, das die Taktgenerierung im Chip übernimmt, am wichtigsten. Bei der MSP430-Familie werden für die verschiedenen Gruppen MSP430F1xx, MSP430P3xx und MSP430F4xx, diverse, innerhalb der Reihe leicht modifizierte Module eingesetzt. In dieser Artikelserie wird speziell auf die MSP430F1xx eingegangen. Informationen zu den Funktionalitäten der anderen Controller-Varianten sind in den einzelnen User Guides [3/4/5], die im Internet [1] zu finden sind, erhältlich.

Ein Blockschaltbild des so genannten

MSP430F14x	
x	Speichergröße
0	1 kByte
1	2 kByte
2	4 kByte
3	8 kByte
4	12 kByte
5	16 kByte
6	24 kByte
7	32 kByte
8	48 kByte
9	60 kByte

Bild 2: Die Zuordnung der Speichergrößen zur Controllerbezeichnung

„Basic Clock Module“ ist in Abbildung 4 dargestellt. Im Diagramm ist zu erkennen, dass in diesem Typ drei Oszillatoren integriert sind. Der RC-Oszillator ist komplett integriert. Es sind also für die Funktion dieses Oszillators keine externen Bauteile notwendig. Das bedeutet, dass der Mikrocontroller auch ohne externe Quarze funktionsfähig ist, indem man ausschließlich den DCO- (= Digital Controlled Oscillator) Takt im Mikrocontroller verwendet. Die Ausgangsfrequenz des DCOs hängt jedoch stark von der Temperatur und der

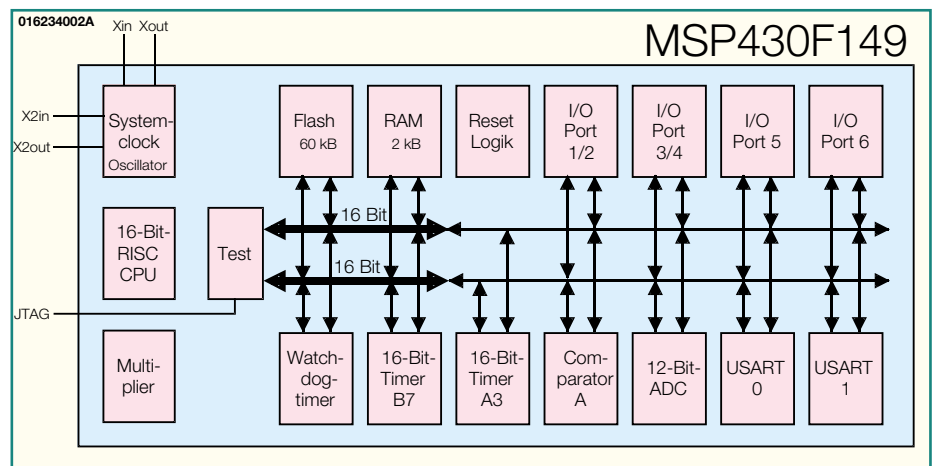


Bild 3: Das Blockschaltbild des MSP430F14x

anliegenden Versorgungsspannung ab. Benötigt man jedoch einen genauen Takt, um beispielsweise eine Zeitmessung durchzuführen, so kann auf die Quarzoszillatoren zurückgegriffen werden. In den Derivaten MSP430F13x und MSP430F14x sind hiervon zwei Stück verfügbar. Bei den Typen MSP430F11x1 und MSP430F12x ist nur ein Quarzoszillator integriert.

Eine Besonderheit stellt noch der LFXT1-Oszillator dar. Dieser kann mit Hilfe des Bits XTS in einen 32-kHz- oder hochfrequenten Mode geschaltet werden. Das heißt, es ist möglich, entweder einen

32-kHz- Quarz oder im hochfrequentem Mode einen Quarz von 450 kHz bis 8 MHz einzusetzen. Im 32-kHz-Mode sind die üblicherweise benötigten Kondensatoren bereits auf dem Chip integriert, sodass nur ein 32-kHz-Quarz anzuschließen ist. Wird jedoch ein hochfrequenter Quarz gewählt, so muss man an beiden Quarzanschlüssen zusätzlich Kondensatoren gegen Masse vorsehen.

Organisation des Speichers

Im Bild 5 ist die Aufteilung des gesamten Adressbereiches eines MSP430F149-Mikrocontrollers dargestellt. Im unteren Bereich des Speichers sind die Control-Register der verschiedenen Peripheriemodule lokalisiert.

Beginnend bei Adresse 0200h befindet sich das RAM. Die Endadresse hängt von der jeweiligen Größe des RAMs in den verschiedenen MSP430-Typen ab. Im Typ MSP430F149 endet das RAM beispielsweise bei Adresse 09FFh.

Eine Besonderheit stellt der Adressbereich 0C00h bis 0FFFh dar. Hier befindet sich ausschließlich in Typen mit Flash-Speicher das „Bootstrap Loader“-Programm. Dieses nicht löschbare Programm ermöglicht eine externe Kommunikation über zwei digitale Ein-/Ausgänge des Chips. Mittels dieser Kommunikation ist

es möglich, den Flash-Speicher des Mikrocontrollers über die serielle Schnittstelle eines PCs zu löschen oder zu programmieren.

Bei den folgenden Bereichen (1000h bis FFFFh) handelt es sich um den Flash-Speicher. Hierbei findet eine Aufteilung in Datenspeicher („Information Memory“) und Programmspeicher („Main Memory“) statt. Prinzipiell ist es aber möglich, sowohl „Main Memory“ als auch „Information Memory“ als Speicher für Programmcode und Daten zu verwenden. Der Flash-Speicher ist byte- oder wortweise program-

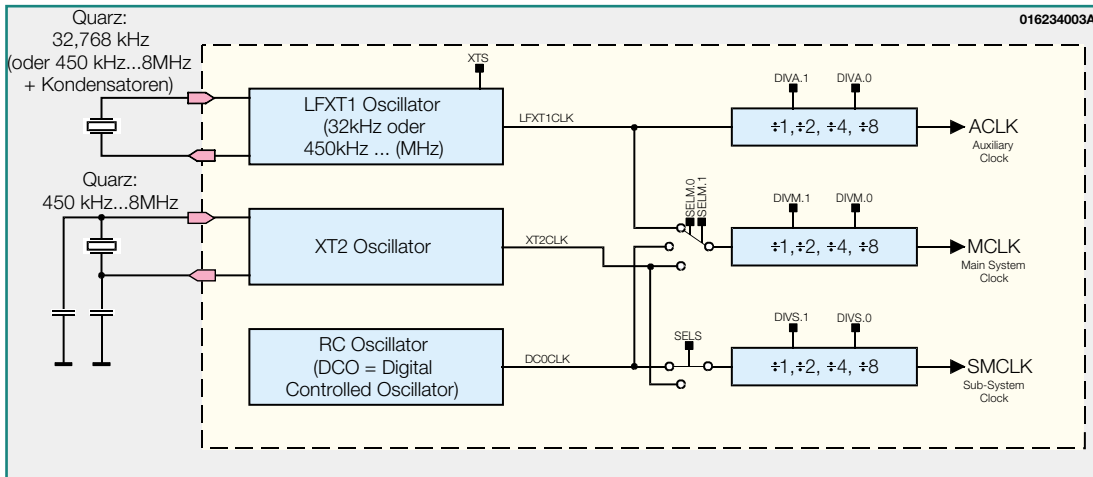


Bild 4: Das Blockschaltbild des „Basic Clock Module“

mierbar, d. h. der ursprüngliche, gelöschte Zustand eines Bits im Flash - dies ist immer der Pegel „1“ - wird auf „0“ gesetzt. Der umgekehrte Vorgang („0“ nach „1“) ist nur segmentweise möglich. Deswegen ist das „Main Memory“ in 512-Byte-Segmente und das „Information Memory“ in 128-Byte-Segmente unterteilt.

Im Adressbereich FFE0h bis FFFFh befindet sich die Interrupt-Vektor-Tabelle. In diesen Adressen werden die einzelnen Startadressen der Interrupt-Service-Routinen definiert. Welcher Interrupt-Vektor dabei zu dem entsprechenden Interrupt gehört, kann man dem jeweiligen Datenblatt entnehmen.

Reset

Nach dem Anlegen der Versorgungsspannung an einem MSP430-Mikrocontroller sollte dieser auch mit der Abarbeitung des im Speicher befindlichen Programmes starten. Hierzu existiert in den Chips eine Reset-Logik. Diese Reset-Logik erkennt bei den MSP430F1xx-Controllern, ob eine Spannung angelegt ist und setzt alle Einstellungen des Chips auf eine

Grundkonfiguration zurück. Diese kann man dem User's Guide [3/4/5] der einzelnen Mikrocontroller entnehmen. Bei der Beschreibung der einzelnen Control-Register wird dort neben der Information, ob die einzelnen Bits lesbar, beschreibbar oder evtl. beides sind, auch angegeben, welchen logischen Pegel das entsprechende Bit nach einem Reset einnimmt.

Um nun ein Reset-Signal zu generieren, ist ein externes Signal an dem Mikrocontroller-Pin RST/NMI notwendig. Die einfachste Beschaltung dieses Pins ist ein Widerstand gegen die Versorgungsspannung des Chips. Zudem ist es empfehlenswert, den RST/NMI-Eingang mit einem Kondensator abzublocken (z. B. 100 nF zwischen RST/NMI-Pin und Vss des Mikrocontrollers). Diese Beschaltung kann nur dann angewandt werden, wenn bei Anlegen der Versorgungsspannung die Spannung beginnend unterhalb von 0,4 V (gemäß Datenblatt) kontinuierlich ansteigt. Ist dies nicht der Fall, so sollte man einen externen SVS- (Supply Voltage Supervisor) Baustein einsetzen.

Watchdog

Jeder MSP430-Mikrocontroller besitzt ein „Watchdog-Timer“-Modul. Dieses Modul ist in zwei verschiedenen Betriebsmodi einsetzbar.

Zunächst kann es als normaler Zähler zum Einsatz kommen, dessen Größe einstellbar ist (6 Bit, 9 Bit, 13 Bit oder 15 Bit). Ein Überlauf dieses Zählers kann einen Interrupt generieren. Diese Funktion wird üblicherweise angewandt, um in der eigenen Anwendung eine Zeitbasis zu definieren. Es wäre zum Beispiel denkbar, dass der Zähler jede Sekunde einen Interrupt generiert und in der Interrupt-Service-Routine durch Software-Zähler die Sekunden, Minuten und Stunden gezählt werden und man somit eine Uhr realisiert.

Neben dem Timer-Betriebsmode ist das Watchdog-Timer-Modul auch im Watchdog-Modus betreibbar. Dies bedeutet, dass

der gewählte Takt den Watchdog-Zähler inkrementiert. Tritt ein Überlauf des Zählers auf, so wird der Mikrocontroller durch einen Reset zurückgesetzt. Der Programmierer muss also dafür sorgen, dass der Watchdog-Zähler zurückgesetzt ist, bevor dieser einen Überlauf generiert. Diese Funktion erlaubt eine gewisse Überwachung der Programmabarbeitung. Hierdurch erkennt man, ob das Programm zum Beispiel einen fehlerhaften Sprung in nicht vorhandene Programmadressen ausgeführt hat.

Anzumerken ist insbesondere, dass das Watchdog-Timer-Modul nach einem Reset immer automatisch als Watchdog konfiguriert ist. Das heißt also, kümmert man sich in seinem eigenen Programm nicht um das Watchdog-Modul, so wird nach Abarbeitung der ersten Zeilen des Programmes ständig der Chip zurückgesetzt. Man sollte sich also bei der Programmierung unbedingt um die Verwendung des Watchdog-Timer-Moduls Gedanken machen. Im einfachsten Fall ist es denkbar, den Watchdog zu deaktivieren.

Nach dieser grundsätzlichen Vorstellung der Mikrocontroller-Familie stellen wir im nächsten Teil der Artikelserie deren Befehlssatz vor. Zudem werden wir erläutern, wie mittels der kostenlos erhältlichen PC-Entwicklungsumgebung „IAR KickStart“ ein Projekt angelegt und ein MSP430-Assembler-Programm geschrieben werden kann. **ELV**

FFFFh	Interrupt-Vektor-Tabelle
FFE0h	
FFDFh	Flash (Programmspeicher)
1100h	
10FFh	Flash (Datenspeicher)
1000h	
0FFFh	Bootstrap Loader
0C00h	
09FFh	RAM
0200h	
01FFh	Peripherie-Control-Register
0000h	

Bild 5: Die Aufteilung des Adressbereiches eines MSP430F149-Mikrocontrollers

- Internet-Adressen:**
- [1] MSP430 Internet Seite: www.ti.com/sc/msp430
 - [2] MSP430 Datenblätter: Link „Product List“ auf MSP430 Internet Seite [1]
 - [3] MSP430x1xx Family User's Guide, SLAU049A
 - [4] MSP430x3xx Family User's Guide, SLAU012
 - [5] MSP430x4xx Family User's Guide, SLAU056A