

Der HC-12 Mikrokontroller

LARS WOLTER , E-MAIL: wolter@inf.fu-berlin.de

Es sind nicht nur die Hochleistungsprozessoren von AMD und INTEL, die in unserer Welt das rechnen übernehmen. Es gibt noch viele andere Prozessoren, die ganz andere Wege gehen. Wir wollen hier eine Kategorie betrachten, die Mikrokontroller. Im Besonderen das neuste Modell der Firma Motorola, welcher viele Funktionen zur Verfügung stellt.

Zuerst betrachten wir Mikrokontroller im Allgemeinen. Anschließend schauen wir uns den Aufbau dieses Mikrokontrollers an. Dann betrachten wir die einzelnen Komponenten etwas genauer, und beschreiben deren Funktionsweise, und auch deren Anwendungsgebiete. Dabei betrachten wir den Kontroller aus der Sicht des Programmierers. Wir werden uns also nicht um den genauen Aufbau in Form von Gattern, oder gar Transistoren kümmern.

Trotzdem ist das Dokument nur für Leute geeignet, die über ein Repertoire an Kenntnissen in Rechnerstrukturen besitzen. Im speziellen sollte die Funktionsweise von Registern und Speicher bekannt sein.

1 Mikrokontroller

1.1 Übersicht

Mikrokontroller sind komplexe Bausteine ähnlich wie Prozessoren. Im Gegensatz zu diesen sind sie jedoch spezialisierter als herkömmliche Mikroprozessoren, wie z.B. die 80x86 die fast jeder in seinem heimischen PC hat. Mikrokontroller werden primär zur Ansteuerung von Gerätschaften verwendet, oder zu deren Überwachung. Diese Aufgabengebiete erfordern selten hohe Rechenleistung sondern eher robuste Interaktionsfähigkeiten. Ein anderer großer unterschied sind die Gebiete in denen sie eingesetzt werden. Ein Mikrokontroller muss auch Geräte unter extremen Bedingungen steuern können, z.B. bei hohen bzw. niedrigen Temperaturen. Dadurch wirken die technischen Daten im Vergleich zu manchen Prozessoren ziemlich jämmerlich, doch ihre Stärken werden schnell sichtbar wenn man genau hinschaut. Denn in diesen Fällen können sie Aufgaben erledigen zu denen ansonsten komplette PC's nötig sind die im Vergleich sperrig und teuer sind.

Ein Mikrokontroller enthält in seinem Chipgehäuse normalerweise nicht nur einen kleinen Prozessor, sondern auch alle Komponenten die für ein komplettes System nötig sind. Sie verfügen über internen Speicher, sowohl Festspeicher als auch Arbeitsspeicher. Außerdem sind die nötigen Komponenten für Schnittstellen integriert, so dass der Chip praktisch alleine ohne eine große Platine auskommt. Es gibt zwar mittlerweile auch komplette PC-Module auf Basis von 486ern oder gar Pentium Systemen, doch sind diese meist sehr teuer, und für Gerätesteuerung selten praktikabel.

1.2 Anwendungsgebiete

In vielen Geräten sind Mikrokontroller enthalten, mit verschiedenen Fähigkeiten. In fast jedem technischen Gerät das man zuhause hat findet man heutzutage einen oder mehrere von diesen kleinen Dingen. Ob das Kühlschränke sind, wo sie die Temperatur regeln, oder im Videorekorder, bei dem sie zur richtigen Zeit die Aufnahme starten. Auch im Auto stecken heutzutage einige von ihnen. Sie kontrollieren Bremsassistenten und Zündvorgänge im Motor. Bei neueren Modellen kontrollieren sie Reifendruck, Scheibenwischer, Klimaanlage, und dienen als Schnittstellen zu einem Bordcomputer (der dann auch im Internet die neusten Staumeldungen abfragt). Selbstverständlich werden sie auch stark in der Industrie eingesetzt. Dort kontrollieren sie Maschinen in Fabriken.

Auch der heimische PC kommt nicht ohne Mikrokontroller aus. Auch wenn die Prozessoren noch so schnell sind, können sie sich nicht um alles kümmern, und so kommen Mikrokontroller in der Peripherie zum Einsatz. Z.B. als Festplattenkontroller oder als Steuerung von Druckern.

Viele von diesen Mikrocontrollern sind sehr spezialisiert, und speziell auf ihre Umgebung zugeschnitten. Es gibt aber auch Controller die flexibler sind und in vielen Gebieten Anwendung finden. Z.B. setzt Das Robocup - Team der Freien Universität Berlin bei ihren Robotern Mikrokontroller ein die die Motoransteuerung übernehmen. Dieses ist der HC-12 den wir noch genauer kennen lernen werden.

2 Der HC-12

2.1 Übersicht

Der HC-12 ist ein 16 Bit Mikrokontroller, der in verschiedenen Ausführungen von der Firma Motorola produziert wird. Die unterschiedlichen Versionen unterscheiden sich in Punkten wie Speicher- ausbau, Schnittstellenkomponenten sowie Gehäuseformen für verschiedene Einsatzgebiete. Eine dieser Versionen der 68HC912DG128 wird von den Robotern der FU Berlin verwendet. Dieser enthält die folgenden Komponenten:

- 16-Bit CPU (CPU12)
- 128K Bytes Flash EEPROM
- 8K Bytes RAM
- 2K Bytes of EEPROM
- Zwei asynchrone serielle Schnittstellen (SCI)
- Ein serielle periphere Schnittstelle (SPI)
- Eine Schnittstelle für IC nach IC Verbindungen (I2C)
- Enhanced capture timer (ECT)
- Zwei CAN 2.0 A,B Module
- Zwei 8-Kanal,10-Bit Analog-Digital Konverter (ATD)

- Ein 4-Kanal Pulse-Weiten-Modulator (PWM)
- Ein lite integration module (LIM) für die Verwaltung von Bus, Interrupts, Timern und anderen Ressourcen.

Diese Version des Chips ist ziemlich gut Ausgestattet, die anderen Versionen verfügen eher über weniger Speicher. Auch in Bezug auf Schnittstellen besitzt dieser alle möglichen Schnittstellen, andere besitzen z.B. ein 8-Kanal PWM, verzichten dann aber auf andere Elemente.

Die Version wird in einem Standardgehäuse geliefert, und ist somit nicht für die extremsten Umgebungen geeignet. Jedoch gibt es praktisch alle Versionen auch in speziellen Gehäusen. Einzige Versionen mit niedriger Stromaufnahme sind auf bestimmte Versionen beschränkt. So gibt es alle 12 Versionen in einer 5 Volt Ausführung, aber für 3,3 Volt gibt es nur eine Version ohne Pulsweitenmodulierung und ohne Flashspeicher.

Die meisten Komponenten des Chips lassen sich abschalten und/oder in einen Stromsparmodus versetzen um die Leistungsaufnahme des Chips zu verringern. Dieses ist recht wichtig, da in vielen Anwendungen der Chip in bereichen eingesetzt wird wo nur ein Akku für die Stromversorgung zur Verfügung steht.

Eine Besonderheit des HC-12 ist seine Möglichkeit komfortables debugging durchzuführen. Dies ermöglicht eine Schnittstelle zum Hintergrunddebuggen. Diese erlaubt es dem Programmierer von einem Host Rechner über ein Debugmodul Das auf dem Chip laufende Programm anzuhalten, Breakpoints zu setzen, und sogar direkt Register und Speicher auf beliebige Werte zu setzen. Dies vereinfacht die Entwicklung mit dem Chip um ein vielfaches.

2.2 Anwendungsgebiete

Bei den FU-Fightern¹ benutzen wir den Chip auf jedem Roboter, damit er die Motoren ansteuert und Funkpakete auswertet. Dabei berechnet die CPU anhand von Daten der Motoren die entsprechenden Motorgeschwindigkeiten. Diese werden dann per Pulsweitenmodulierung in Steuersignale für die Motoren umgesetzt. Die CPU übernimmt es auch Pakete von einer der Seriellen Schnittstellen zu übernehmen und diese auszuwerten. Dabei wird auch eine Fehlerkorrektur durchgeführt. Zu Debugzwecken kann dann auch noch die zweite Schnittstelle verwendet werden um Daten vom Roboter zurück zu einem externen Rechner zu schicken. Früher benutzten wir eine Vorgängervariante des HC-12, den HC-05. Dieser war nur mit einer 8 Bit CPU ausgestattet, besaß nur 256 Byte Programmspeicher und hatte kaum weitere Komponenten zur Kommunikation zur Verfügung. Das wir dieses Modell angeschafft hatten ist allerdings auch erst zwei Jahre her. Das zeigt dass auch in diesem Bereich die Entwicklung und Miniaturisierung stetig voranschreitet.

¹ Die FU-Fighters sind das Robocup Team der Freien Universität Berlin (<http://www.inf.fu-berlin.de>) . Das Team existiert mittlerweile seit drei Jahren und hält seit dieser Zeit den Vize-Weltmeistertitel im Robocup. Für weitere Informationen besuchen Sie doch einfach unsere Webseite unter <http://www.fu-fighters.de>.

Die einzigen anderen Anwendungsgebiete des HC-12 die ich bis jetzt finden konnte sind Universitätsprojekte wo sie zur Ansteuerung von diversen Gerätschaften verwendet werden. Dieses liegt wohl an der relativen Neuheit des Chips.

2.3 Aufbau

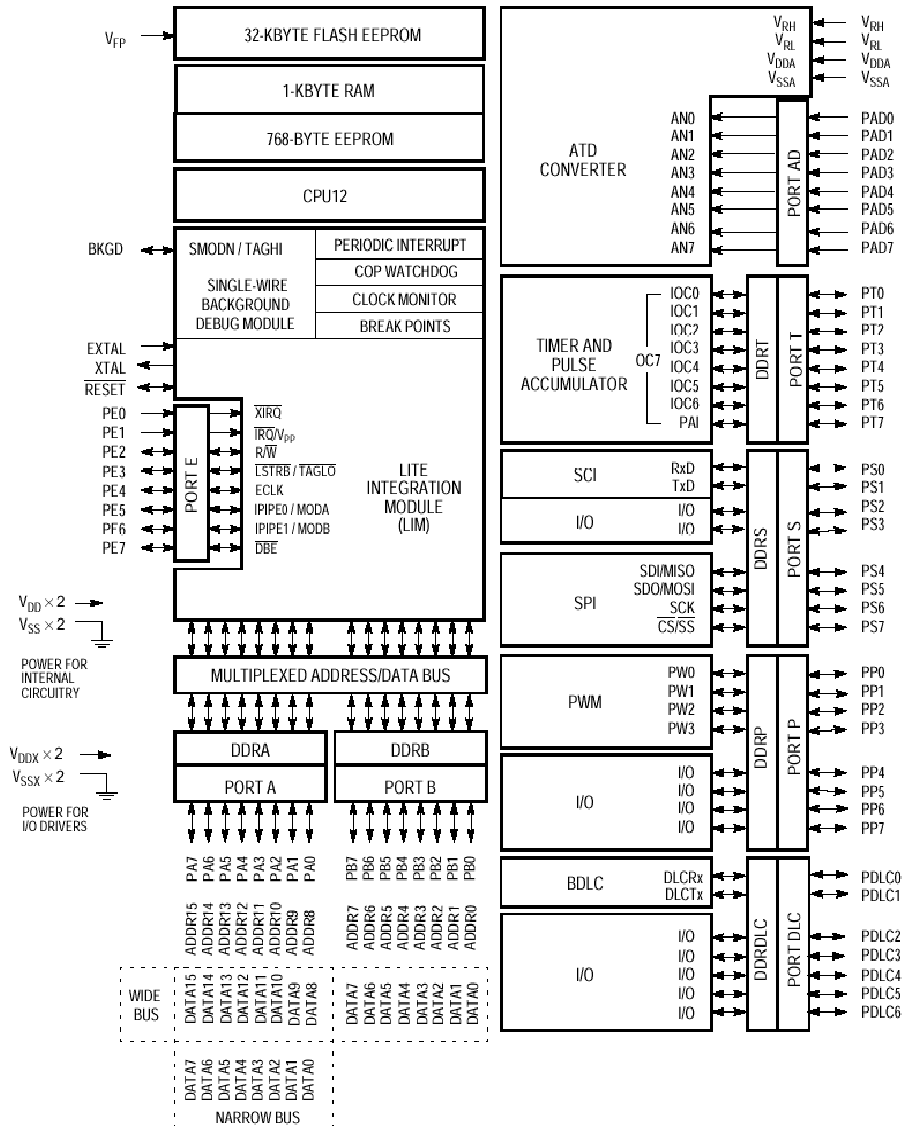


Abbildung 1: Aufbau des HC-12 Mikrokontrollers

Das Gehäuse des Kontrollers verfügt über 112 Pins um mit der Außenwelt zu kommunizieren. In dem Diagramm kann man gut die Verschiedenen Komponenten sehen, und vor allem erkennen, dass die CPU einen sehr kleinen Teil in dem Chip einnimmt. Dies liegt zum einen an den vielen Komponenten über die dieser Chip zusätzlich verfügt. Aber auch die breite Schnittstelle nach außen nimmt sehr viel Platz ein.

Komponenten wie Speicher hingegen sind direkt über den 16 Bit breiten Bus des Chips miteinander verbunden und benötigen keine Schnittstelle nach außen.

3 Die CPU 12

3.1 Übersicht

Die 16 Bit CPU des HC-12 beruht sehr stark auf ihren Vorgängern, die noch 8bittig waren. Daraus resultiert auch der Aufbau der Register, die noch immer auf 8 Bit Operationen ausgelegt sind. Auch wenn der Chip 16 Bit Operationen durchführen kann. Hauptsächlich ist die Erweiterung seines Adressraumes auf 16 Bit der größte Schritt gegenüber den vorigen 8 Bit Versionen. Dadurch verfügt der Chip über sehr viel Speicher für Programme und Daten, die direkt verwendet werden können. Die CPU funktioniert wie auch jede andere, die mit Registern arbeitet. Jedoch ist die Anzahl der Register sehr gering, und ihre Verwendung eingeschränkt. Die Anzahl an verfügbaren Operationen kann auch nicht mit denen moderner CISC Architekturen konkurrieren, ist jedoch für die Anwendungsbereiche durchaus angemessen.

3.2 Register

Wie bereits erwähnt, ist der Registerumfang der CPU12 nicht sehr groß. Die Arbeitsregister sind zwei 8 Bit Register, die so genannten Accumulatoren. Diese können für 16 Bit Operationen zu einem Register zusammengefasst werden. Die beiden Index Register sind hauptsächlich für indirekte Speicheradressierung, können aber auch für Rechenoperationen mit den Accumulationsregistern missbraucht werden. Jedoch sind die Operationen auf diesen Registern beschränkt. Die Register für Stackpointer und Programm Counter sollten sich von selbst erklären. Der Stackpointer zeigt auf die letzte Position im Stack. Der Programcounter zeigt auf den nächsten Befehl. Dieser Aufbau repräsentiert die äußere Sichtweise, wie sie der Programmierer sieht. Intern werden jedoch breitere 20 Bit Register für Rechenoperationen verwendet. Dadurch erhält man eine erhöhte Präzision bei aufeinander folgenden Rechnungen ohne Speicherzugriffe. Die CPU unterstützt mehrere Datentypen in den Registern:

- Bit data
- 8-Bit und 16-Bit signed und unsigned Integers
- 16-Bit unsigned Kommazahlen
- 16-Bit Adressen

Intern verwendet der HC-12 Register mit einer höheren Auflösung. Diese sind dann 20-Bit breit. Und auch die ALU rechnet mit diesen 20-Bit Werten. Dies führt zu einer höheren Genauigkeit bei Operationen, die komplett in der CPU und ihren Registern stattfinden. Erst wenn Daten in den Speicher kopiert werden, geht Auflösung verloren.

3.3 Speicheradressierung

Die CPU kann auf vielfältige Art und Weise auf ihren Speicher zugreifen.

Inherent -- Alle Operanden sind in den CPU Registern vorhanden, das heißt es sind keinerlei Speicherzugriffe nötig.

Immediate -- Operanden sind Teil der Anweisung. Das tritt auf wenn z.B. bei einer Addition der eine Operand in der Anweisung enthalten ist, während der andere Operand bereits im Register ist.

Direct -- Operand ist Teil der Adresse. Dieser Modus ist nötig beim Berechnen von Adressen, in diesem Fall wird zum Beispiel nur ein Offset auf eine Adresse addiert.

Relative -- Mit Offset zum Programm Counter. Diese Adressierungsart ist nur bei Programmsprüngen sinnvoll, um relativ im Programm zu springen. Also z.B. um die folgende Anweisung zu überspringen.

Diverse **Indizierte Adressierungsarten** vervollständigen diese Liste. Bei diesen gibt es mehrere Varianten, die automatisch inkrementieren oder dekrementieren.

4 Zeitsignale

4.1 Übersicht

Zeitsignale werden in vielen Situationen benötigt. Und es ist sehr wichtig unterschiedliche Signale zur Verfügung zu stellen. Viele externe Geräte erwarten ein Signal in bestimmten Zeitintervall. Dazu reicht es nicht sich allein auf die Taktrate des Prozessors zu verlassen, da diese je nach Bauart und Version unterschiedlich sein kann. Außerdem benötigt man evtl. ein Signal welches nicht ein Vielfaches der Taktrate ist. Hierfür sind Zeitsignale, im Englischen Timer genannt, vorhanden. Diese ändern je nach Einstellung bestimmte Register, oder sorgen für die Auslösung eines Interrupts.

Der HC-12 verfügt über mehrere interne Timer, sowie die Möglichkeit Taktsignale von außen einzuspeisen.

4.2 Die verschiedenen Signale

Der phase-locked loop (PLL) des 68HC912DG128 kann Frequenzen von 0,5 bis 8 MHz erzeugen. Dabei stehen nicht alle Signale des HC-12 jedem Bauteil des Chips zur Verfügung.

- **TCLK1 bis TCLK4**
Diese 4 Signale stehen der CPU zur Verfügung
- **ECLK und PCLK Signale**
Diese beiden sind für Schnittstellen Inter-IC Bus, serielle periphere Schnittstelle, Pulsweitenmodulatoren und die beiden Analog nach Digital Konverter.
- **MCLK Signal**
Dieses ist für die seriellen Schnittstellen und den Enhanced Capture Timer

- XCLK Signal
Dieses wird vom RTI und COP verwendet
- SCLK Signal
Zur Kalibrierung von äußeren Signalen
- CANCLK
Der CAN-Bus besitzt ebenfalls ein eigenes Zeitsignal

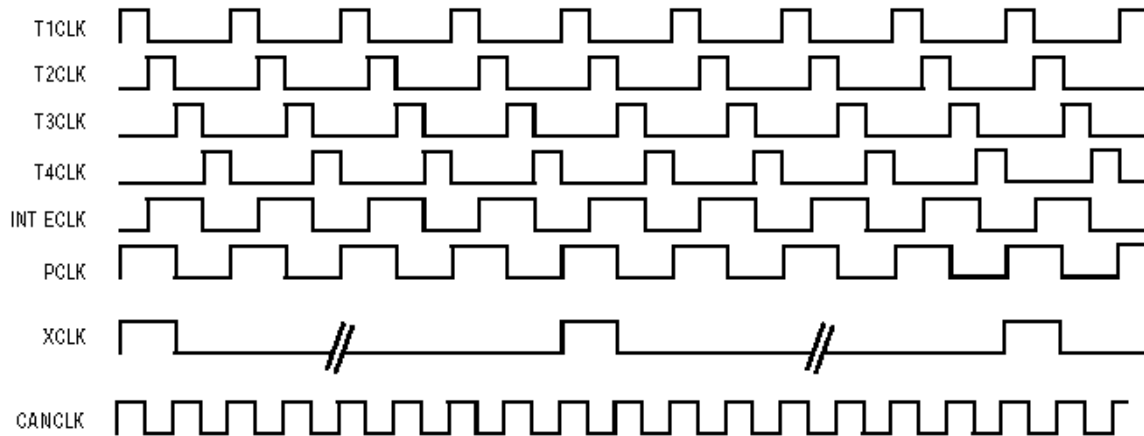


Abbildung 2 Verhältnis zwischen den Zeitsignalen

4.3 Der Watchdog

Der Computer Operating Properly (COP) Timer wird auch Watchdog genannt. Dieser sorgt dafür das sich der Controller nicht aufhängt. Wenn man diesen Sicherheitsmechanismus verwendet, muss das Anwendungsprogramm vor Ablauf einer gewissen Zeit in ein bestimmtes Register zwei Werte schreiben. Dieses soll signalisieren, das das Programm noch immer normal läuft, und nicht z.B. in einer endlos Schleife fest hängt. Wird das Register nicht beschrieben führt der Controller automatisch einen Reset durch. Im Gegensatz zu einem Desktop PC ist das auch nicht so Problematisch, da der Bootvorgang des Controllers lediglich einige Millisekunden in Anspruch nimmt.

Für größere Flexibilität lässt sich der COP Timer so konfigurieren, dass er dem Programm verschiedene lange Zeit lässt um den Reset zu verhindern. Außerdem gibt es auch eine Option die dazuführt, dass verfrühtes schreiben in das Register zu einem Reset führt.

5 Schnittstellen

5.1 Serielle Schnittstellen

Diese Version des HC-12 ist mit zwei seriellen Schnittstellen ausgestattet wie man sie auch vom PC kennt. Diese beiden Schnittstellen erlauben die Kommunikation mit externen Geräten, die ebenfalls eine RS-232 kompatible Schnittstelle besitzen. Die Schnittstellen unterstützen nur ein Kommunikationsformat mit einem Startbit, 8 oder 9 Datenbits und einem Stopbit. Die Baudrate der Schnittstelle kann frei gewählt werden, in einem Bereich von 110 Baud bis 38400 Baud.

Wird das als Übertragungsformat die Variante mit acht Datenbits gewählt, so fungiert das neunte Bit als Paritäts Bit zur Fehlerkorrektur. Dabei wird die Berechnung des Bits von der Hardware übernommen.

Als zusätzliche Features besitzt die Schnittstelle die Möglichkeit zu erkennen wann Daten ankommen, und bzw. ob gerade Daten übertragen werden.

110
300
600
1200
2400
4800
9600
14400
19200
38400

Schließlich gibt es noch eine dritte serielle Schnittstelle. Diese serielle periphere Schnittstelle dient zur synchronen Kommunikation mit anderen Controllern oder Geräten. Dabei verwendet die Schnittstelle das Master - Slave² Prinzip (siehe unter IIC). Auch Verbindungen mit mehreren Geräten die Master sind werden unterstützt. Zum Austausch von Daten stehen zwei Register zur Verfügung, ein Master und ein Slave Register. Beim Austausch der Daten werden automatisch die Daten des Masters gesendet, während die Daten des anderen im Slave landen, je nach Modus.

Tabelle 1 Baudraten

5.2 Inter-IC Bus (IIC oder I²C)

Der IIC ist ein zwei Adrigger, serieller Bus für Transfers in beide Richtungen. Er ist eine effiziente und einfache Methode um mit anderen Chips zu kommunizieren. Dabei bietet er eine maximale Übertragungsrate von 100kbps. Auch dieses System verwendet eine Slave – Master Technik. Das heißt, dass einer der Chips die an der Verbindung teilhaben ein Master ist. Dieser Master hat die Kontrolle über die Slaves. Er kann also sowohl Daten von einem Slave anfordern, als auch Daten an einen Slave senden. Die Slaves hingegen sind passiv, und warten auf eine Aktion des Masters um zu reagieren. Der IIC unterstützt auch Verbindungen, an denen mehrere Master beteiligt sind.

5.3 Digitale Ein- und Ausgänge

Es gibt insgesamt 84 Ports, von denen 66 als Ein und Ausgänge benutzt werden können, und 18 nur als Eingänge. 16 Der Digitalen Ports unterstützen so genannte Key Wake Ups. Diese sorgen dafür dass die CPU aufgeweckt wird wenn sich an einem der Pins etwas tut.

² Slave - Sklave, beschreibt den passiven Teil, während der Master die aktive Kontrolle hat

5.4 Analog nach Digital Konverter (ADT)

Diese dürften jedem aus dem Audiobereich bekannt sein. Doch man kann ein ADT nicht nur zum digitalisieren von Musik verwenden. Dieser HC-12 besitzt zwei 8 Kanal ADT's mit je 10 Bit Auflösung. Bei der Samplerate der Wandler kann man zwischen 2,4,8 und 16 Zeitsignalen wählen. Auch die Auflösung kann man auf Wunsch auf 8 Bit ändern.

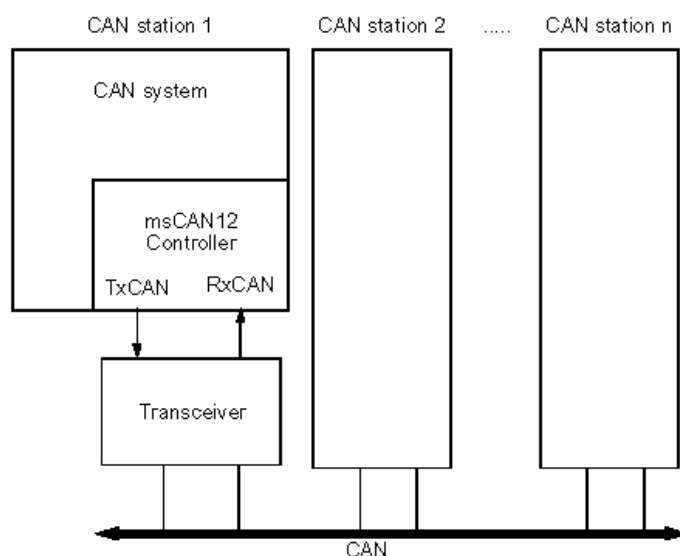
Meistens werden diese zum Auswerten von analogen Messgeräten verwendet, da die Meisten Sensoren analoge Daten liefern, wie z.B. Temperatursensoren, Drucksensoren oder eben Mikrofone.

5.5 Puls Weiten Modulation (PWM)

Es gibt vier PWMs auf dem Chip,. Jeder der PWMs bietet eine Auflösung von 8 Bit. Falls man eine höhere Auflösung benötigt, kann man je zwei der Kanäle zusammenlegen, um eine Auflösung von 16 Bit zu erhalten. Die PWMs werden z.B. zur Ansteuerung von Elektromotoren verwendet. Dabei regelt man die Geschwindigkeit des Motors indem man ihm Impulse verschiedener Länge schickt. Je länger die Impulse sind desto schneller dreht sich der Motor. Dadurch lassen sich Motoren sehr präzise steuern. Um die Modulationseinheiten zuprogrammieren, setzt man zuerst fest wie lang der Abstand zwischen den Impulsen ist. Anschließend schreibt man Daten in die Register der PWMs um die Impulsfolgen zu erzeugen. Dabei sind die Register gepuffert, so dass man auf entweder das Signal bekommt was man vor dem schreiben hatte oder gleich das neue, aber kein Signal das irgendwie dazwischen liegt.

5.6 CAN Bus

Der CAN Bus ist ein System, das entwickelt wurde für Fahrzeugkomponenten, die miteinander kommunizieren sollten. Dabei lagen die Schwerpunkte auf Echtzeitverarbeitung, Betriebssicherheit und niedrigen Kosten. Die eigentlichen CAN Module sind über einen Transceiver an den Bus angebunden. Dieser sorgt dafür, dass defekte oder deaktivierte Module den CAN Bus nicht beeinträchtigen. Das CAN Modul kann sich entweder in einem dominanten oder zurückhaltenden Zustand befinden. Die eigentlichen Daten werden in Forma von Nachrichten über den Bus verschickt, wobei jede Nachricht eine entsprechende Priorität besitzt. Die Nachrichten besitzen eine Größe von 13 Bytes, wobei die Nachricht eine Größe von 13 Bytes, wobei der Datenteil der Nachricht lediglich 8 Bytes groß ist. Eigene Nach-



Nachrichten werden von dem Modul zwar empfangen, aber ignoriert. Wird das Modul am CAN Bus aktiv, kann von Programmseite die Konfiguration des Moduls nicht mehr geändert werden. Das dient dazu Fehler im CAN Bus zu vermeiden die durch Programmfehler entstehen könnten.

Diese Version des HC-12 besitzt zwei identische CAN Module, die das CAN 2.0 A/B Protokoll welches in der BOSCH Spezifikation vom September 1991 angegeben ist implementieren. Diese Module heißen msCAN, was für „motorola scalable CAN“ steht.

6 Speicher

Der MC68HC912DG128 besitzt 128 K FLASH EEPROM. Dieser Speicher dient zum ablegen von Programmcode und Daten auf die schnell zugegriffen werden muss. Die Anbindung geschieht über einen 16 Bit breiten Bus, so dass einzelne Bytes und auch Words in einem Clock Zyklus gelesen werden können. Zur Programmierung dieses nichtflüchtigen Speichers benötigt man eine höhere Spannung, als zum normalen Lesebetrieb. Diese muss extern zugeführt werden. Der Flashspeicher kann von der CPU zu Initialisierung benutzt werden, indem in ihm ein Bootblock eingerichtet wird.

Die 2 K EEPROM sind ebenfalls für Programmcode, sie benötigen zwar ebenfalls eine höhere Spannung zum löschen und neu programmieren, allerdings wird diese intern zur Verfügung gestellt.

Als Arbeitsspeicher fungieren die 8 K RAM die während des Programmablaufs beliebig genutzt werden können.

7 Literatur und Links

Der Großteil der Informationen stammt von Motorola (<http://www.e-motorola.com>) speziell aus dem PDF Dokument: *68HC912DG128 Advance Information Rev 2.0 (2001)*

Andere Informationen beruhen auf praktische Erfahrungen beim arbeiten mit dem Chip bei den FU-Fightern (<http://www.fu-fighters.de>)