



# **Technische Dokumentation**

**für die Elektronikbaugruppe**

**Soundmodul SDM**

**Version 1.0**

Dateiname: BA23-SDM.ZIP   Ausgabe 01.03.2018

## Inhaltsverzeichnis

	Seite
<b>1 Beschreibung der Baugruppe -----</b>	<b>3</b>
1.1 Allgemeines -----	3
1.2 Schaltungsbeschreibung -----	4
1.3 Speichermedium -----	4
1.4 ELV Player -----	4
<b>2 Programmierung -----</b>	<b>5</b>
2.1 Entwicklungsumgebung -----	5
2.2 Programmcode und Programmierung -----	5
<b>3 Ausführung -----</b>	<b>6</b>
3.1 Beschreibung der Anschlüsse -----	6
<b>4 Allgemeines -----</b>	<b>7</b>
4.1 Literaturverzeichnis -----	7
4.2 Dateienverzeichnis -----	8
4.3 Elektronische Bauelemente -----	8
4.4 Hinweise zu den Nutzungsrechten -----	8
4.5 Technische Hinweise -----	10
4.5.1 Ausdrucken der Dateien -----	10
4.5.2 Gehäuse und Frontplattenbeschriftungen -----	10
4.5.3 Bauelementeauswahl -----	10
4.5.4 SMD-Bestückung -----	10

# 1 Beschreibung der Baugruppe

## 1.1 Allgemeines

**Das wohl bei seiner Größe leistungsfähigste Soundmodul, was derzeit von Hobbyelektronikern aufgebaut werden kann, soll neben der Vorstellung der technischen Raffinesse ebenso als Einstieg für diejenigen gedacht sein, der bisher um Leiterplatten mit SMD-Bestückung oder der C-Programmierung einen großen Bogen gemacht hat.**

**Der japanische Softwareentwickler Chan [1] hat dafür mit seinem Open-Source-Project zur Anbindung von microSD-Karten an die Mikrocontroller der AVR-Familie die Grundlagen bereitgestellt - dafür an dieser Stelle herzlichen Dank!**

Bei der Recherche nach einem leistungsfähigen Geräuschgenerator für den Modellbau stößt man im Internet auf eine Reihe interessanter Lösungsansätze. Allerdings wird es erst durch die Anbindung von Speichermedien möglich, Sounddateien größerer Länge und akzeptabler Qualität wiederzugeben. Dabei stellen die zeitkritische Bereitstellung des Datenstromes und die Dateiverwaltung eine große Herausforderung an die Elektronik dar, die durch den Entwickler in hervorragender Weise umgesetzt wurde.

Mit den „Petit FAT File System Module“ [2] ist es bereits mit 8bit-Mikrocontrollern der ATtiny-Serie, die nur einen relativ kleinen RAM-Speicher zur Verfügung stellen, möglich, ein Dateiverwaltungssystem einzusetzen, was eine entscheidende Grundlage für den Aufbau des vorgestellten Moduls bedeutet.

Bereits die 8-Pin Mikrocontroller der ATtinyX5-Serie verfügen über zwei schnelle Ausgangskanäle (PWM). Mit einer Frequenz von 250 kHz ermöglicht dies bereits die Bereitstellung eines ausreichenden Frequenzbereiches für die Ausgabe eines Audio-Signals (Klangsynthese durch Frequenzmodulation [3]).

Mit dem Player ist es nur möglich, Dateien im WAVE/RIFF-Format abzuspielen (Microsoft Wave-Datei im LPCM-Format, 8/16-bit; Mono/Stereo bis 48 kHz Abtastrate; jede Konfiguration akzeptiert diesen Bereich).

Andere Formate, wie z. B. das MP3-Format, müssen zunächst gewandelt werden. Im Internet findet man dazu entsprechende Konverter.

Die Dateien, die fortlaufend dreistellig nummeriert werden müssen (001.wav, 002.wav, ...), können direkt im Wurzelverzeichnis (Root) abgelegt werden.

Auf der Homepage des Entwicklers werden 4 Schaltungsvorschläge für die ATtiny 25/45/85 und einer für den ATtiny861 aufgeführt:

1. Mono (ATtiny 25/45/85)
  - Signalausgang über Koppelkondensator
  - ISP-Programmierung
2. Stereo (ATtiny 25/45/85)
  - Signalausgang über Koppelkondensator
  - ISP-Programmierung nur einmalig möglich, da das Reset-Pin schaltungstechnisch benötigt wird; ansonsten nur noch programmierbar mittels HVSP
3. Mono-16 bit (ATtiny 25/45/85)
  - Signalausgang über Koppelkondensator
  - ISP-Programmierung nur einmalig möglich, da das Reset-Pin schaltungstechnisch benötigt wird; ansonsten nur noch programmierbar mittels HVSP
4. Mono-OCL (ATtiny 25/45/85)
  - Signalausgang ohne Koppelkondensator
  - ISP-Programmierung nur einmalig möglich, da das Reset-Pin schaltungstechnisch benötigt wird; ansonsten nur noch programmierbar mittels HVSP
5. optional (ATtiny 861)
  - Sounddatei über Auswahlstern abrufbar
  - ISP-Programmierung

## High Voltage Serial Programming (HVSP)

Mit einer kleinen Schaltung nach [4] können bei Bedarf die eingesetzten AVR's auf ihre Werkseinstellungen zurückgesetzt werden.

Hinweis

- Reduktion des Stromverbrauches durch Einsatz einer Serieninduktivität im PWM-Ausgang
- ggf. Einsatz eines Tiefpasses im Audioausgang (low-pass filter: LPF, z. B. nach [10])
- keine direkte Anschaltung des PWM-Ausgangs an einen externen Verstärker (Zerstörungsgefahr)

## 1.2 Schaltungsbeschreibung

Um Interessenten einen Einstieg in das Experimentieren mit dieser Art der Sounderzeugung zu vereinfachen, wurde der erste Schaltungsvorschlag etwas modifiziert und eine Leiterplatte mit SMD-Bestückung entworfen. Um die manuelle Bestückung etwas zu erleichtern, wurden Bauelemente der Bauform 1206 vorgesehen.

Da microSD-Karten (MSDK) nur an einer Spannung von ca. 3 V betrieben werden dürfen, wurde ein Festspannungsregler auf der Leiterplatte integriert. Ein Wannenstecker, der je nach Einsatzfall seitlich oder frontal eingelötet werden kann, ermöglicht den komfortablen Anschluss an ein Programmiergerät (SPI). Über ihn kann ebenfalls das NF-Signal abgenommen und je nach Bedarf an einen NF-Verstärker weitergeführt werden.

Bei einem ausschließlichen Batteriebetrieb des Moduls ist die Bestückung der 3,3V-Spannungsversorgung (Festspannungsregler) nicht erforderlich.

Die Leiterplatte ist für eine Bestückung mit einem, u. a. in Handys eingesetzten microSD-Sockel mit Klappmechanismus ausgelegt, da dieser Sockeltyp zur Zeit der Manuskripterstellung preiswert verfügbar war.

Leiterplatte und Sockel können bei Interesse über den Autor bezogen werden.

Falls die erreichbare Lautstärke mit einem direkt angeschlossenen Lautsprecher nicht ausreicht, stellt es kein Problem dar, die Schaltung um einen NF-Verstärker zu erweitern.

So könnte zum Beispiel der Schaltungsvorschlag aus dem Datenblatt des TBA820M (SGS-THOMSON MICROELECTRONICS: 1,2 W Audio Verstärker) zur Anwendung kommen, der bereits mit einer Betriebsspannung von 3 V auskommt

## 1.3 Speichermedium

Die MSDK wird am SPI-Bus betrieben. In der Literatur wurde angemerkt, dass nicht alle Speicherkarten darüber ansprechbar sind bzw. sich nicht die erforderliche Datenstromgeschwindigkeit realisieren lässt; die beiden vom Autor eingesetzten MSDK funktionierten ohne Probleme.

Zu beachten ist, dass nur Speicherkarten mit einem FAT12/16/32-System angesprochen werden können. Speicherkarten mit dem extFAT-System (microSDXC) können nicht eingesetzt werden und müssen ggf. umformatiert werden.

## 1.4 ELV Player

Wer gerne einen Komplettbausatz verwenden möchte, kann auf ein Modul des Elektronik-Anbieters ELV zurückgreifen.

Dieser Hersteller hat ebenfalls das Konzept des Japaners Chan aufgegriffen und einen WAVE-Player entwickelt, dessen SMD-Bauteile bereits herstellerseitig bestückt sind.

Bei dieser Variante sind die Wave-Dateien über Taster direkt auswählbar und damit in ihrer Anzahl begrenzt.

Er trägt die Bezeichnung „Mini-Wave-Player - MWP1“, wurde im ELVjournal 03/2013 vorgestellt und kostet aktuell 13,95 €.

Ein kostenpflichtiger Download des Fachbeitrages ist über die Herstellerseite im Internet [13] möglich.

## 2 Programmierung

### 2.1 Entwicklungsumgebung

Für eine ggf. vorzunehmende Modifizierung des in C vorliegenden Quellcodes bietet sich WinAVR [5] an. Nach dem Download und der Installation des open-source Programmpaketes kann ein neues Projekt erstellt werden, dem die Quelldateien für das Soundmodul zugeordnet werden.

Allerdings bietet es sich zunächst an, den integrierten Editor ggf. durch die aktuelle Version [7] zu ersetzen.

Um in diesem Fall die Erzeugung des Hexcodes einzuleiten, ist es erforderlich, das Menü nach der Installation anzupassen, um das „Makefile“ aufrufen zu können, was das Compilieren einleitet.

Die Anleitung dazu findet sich in [6] im Unterpunkt 5. Daneben sind detaillierte Hinweise zur gesamten Installation aufgeführt.

Diese Anleitung ist allerdings in Englisch; eine Alternative für die Begleitung der ersten Schritte bietet sich mit der Diplomarbeit von Bianca-Charlotte Liehr [8].

### 2.2 Programmcode und Programmierung

Im Quellcode hat der Entwickler [1] für die Schaltungen mit den ATtiny 25/45/85 verschiedene Makefiles angegeben, die entsprechend der aufgebauten Schaltung für die Code-Generierung zur Anwendung kommen.

Mit dem C-Compiler wird ein File (Hexcode) erzeugt, das am Ende des Codes einen Bereich beinhaltet, der für die Einstellungen der FUSE-Bits des MC relevant ist.

Da nicht alle Brenner dafür ausgelegt sind, diese Angaben beim Programmieren in den MC zu übertragen (FUSE-Bits einstellen), ist ggf. dieser Bereich vor dem Beschreiben manuell mit einem Editor (Bild: Fusebereich im HEX-Code.jpg) zu entfernen und es müssen die FUSE-Bits manuell eingestellt werden.

Der MC in SMD-Bauform muss nicht unbedingt vor dem Einbau programmiert werden, da alle für eine SPI-Programmierung erforderlichen Anschlüsse am Steckverbinder anliegen.

Nach dem Zuschalten der Betriebsspannung oder dem Einlegen der MSDK werden in der Softwareversion von Chan die Wave-Dateien der Reihe nach abgespielt. Mit dem Taster ist es möglich, während des Abspielens direkt zur nächsten Datei zu springen.

Die leicht vom Autor modifizierte Version wartet nach dem Abspielen der ersten Datei solange, bis ein Tastendruck erfolgt.

Da es sich um Freeware handelt und der Quelltext offen liegt, kann sich jeder die Software nach den individuellen Vorstellungen anpassen.

## 3 Ausführung

### 3.1 Beschreibung der Anschlüsse

Folgende Übersicht erleichtert den Anschluss:

X1	Anschluss	Programmierung	Betrieb
X1: 1	MOSI	AVR-Programmiersignal	
X1: 2	SV+	-	Eingangsspannung $\leq 12\text{ V}$
X1: 3	3,3V+	-	(Betriebsspannung 3,3 V)*
X1: 4	TAS	-	Anschluss Taster (extern)
X1: 5	/RES	AVR-Programmiersignal	
X1: 6	AUDIO	-	Lautsprecher (Ausgang)
X1: 7	SCK	AVR-Programmiersignal	
X1: 8	GND	Masse	
X1: 9	MISO	AVR-Programmiersignal	
X1:10	GND	Masse	Lautsprecher (Masse)

\*) Bei einem Direktanschluss von ca. 3 V (z. B. 2x AAA) ist der Anschluss über Pin 2 nicht erforderlich; er kann ggf. für einen zusätzlichen NF-Verstärker verwendet werden.

## 4 Allgemeines

### 4.1 Literatur- und Quellenverzeichnis

#### Homepage

- [1] Homepage des Japaners Chan  
<http://elm-chan.org>
- [2] Petit FAT File System Module  
[http://elm-chan.org/fsw/ff/00index\\_p.html](http://elm-chan.org/fsw/ff/00index_p.html)

#### Allgemeines

- [3] Klangsynthese durch Frequenzmodulation  
<https://www2.ak.tu-berlin.de/~fhein/Alias/Studio/FM/>
- [4] Fuse Resetter per HVSP  
<http://homepage.hispeed.ch/peterfleury/avr-hvsp-fuse-restore.html>
- [5] WinAVR  
Eine Distribution des Cross-Compilers „avr-gcc“ zum Einsatz unter Windows  
<http://winavr.sourceforge.net/>
- [6] Downloading, Installing and Configuring WinAVR  
Punkt 5.0: Setting up Programmers Notepad  
[http://winavr.sourceforge.net/install\\_config\\_WinAVR.pdf](http://winavr.sourceforge.net/install_config_WinAVR.pdf)
- [7] Programmer's Notepad  
<http://www.pnotepad.org/>
- [8] Einführung in die Programmierung des AVR-Controllers  
[http://www.physik.uni-regensburg.de/studium/edverg/elfort/PDFs/Diplomarbeit\\_BLiehr.pdf](http://www.physik.uni-regensburg.de/studium/edverg/elfort/PDFs/Diplomarbeit_BLiehr.pdf)
- [9] Fuse-Calc  
<http://www.engbedded.com/fusecalc>
- [10] Datenblätter Mikrocontroller (Fa. Microchip)  
<http://www.microchip.com/design-centers/8-bit>
- [11] Audio Filter  
<http://www.elektronik-labor.de/AVR/T13contest/sdsound-tn13.htm>
- [12] Lizenz (Deutsch)  
<http://creativecommons.org/licenses/by/3.0/deed.de>
- [13] ELV: Mini-Wave-Player - MWP1, Komplettbausatz  
<http://www.elv.de/mini-wave-player-mwp1-bausatz.html>  
[http://www.elv.de/Sound-direkt-vom-Chip-%E2%80%93-Mini-Wave-Player-MWP1/x.aspx/cid\\_726/detail\\_42440](http://www.elv.de/Sound-direkt-vom-Chip-%E2%80%93-Mini-Wave-Player-MWP1/x.aspx/cid_726/detail_42440)

## 4.2 Dateienverzeichnis

Für das Arbeiten mit der Dokumentation können der kostenfreie Adobe- oder Foxit-Reader verwendet werden, da alle Dateien im \*.pdf Format veröffentlicht sind.

Für das Öffnen der Target-Datei (T3001-Format) kann die kostenfrei angebotene Version vom Leiterplattenhersteller Beta-LAYOUT (<https://de.beta-layout.com/leiterplatten/technik/downloads/>) eingesetzt werden.

### Dateien der Dokumentation

01. SDM (001 Kurzinformation).pdf	Vorstellung der Elektronikbaugruppe
02. SDM (002 Dokumentation).pdf	Dokumentation (geöffnet)
03. SDM (003 Fuse-Lock-Bits).pdf	Einstellung für MC
04. SDM (101 Stromlaufplan).pdf	
05. SDM (V x).T3001	Leiterplatte (kpl.); x: aktuelle Version

### Hinweis

Folgende Dateien können bei Bedarf aus der beiliegenden Target-Datei generiert werden:

- Stromlaufplan
- Bestückungsplan
- Löt- und Bestückungsseite der Leiterplatte

### Programmbeispiel (Quelltext)

10. SDM sd8psrc (Original, Chan).zip
11. SDM sd8psrc (Modifikation).zip

### Hinweis

Die Angaben zu den Fuse- und Lock-Bits sind zu beachten.

## 4.3 Elektronische Bauelemente

Für Elektronikamateure, die in Ihrer Nähe keinen Fachhandel für elektronische Bauteile haben oder denen die Bestellung, zum Beispiel beim Elektronikversand Reichelt (<http://www.reichelt.de/>) für ein oder zwei Fehlteilen zu aufwendig sein sollte, können wir (nur innerhalb von Deutschland) eventuell Hilfe bei der Beschaffung anbieten.

Das Angebot an Leiterplatten ist abhängig von der Nachfrage und der jeweiligen Baugruppe. Die Leiterplatten sind ein- oder zweiseitig, gebohrt und beschichtet; zweiseitige seit mehreren Jahren durchkontaktiert.

## 4.4 Hinweise zu den Nutzungsrechten

Die Bauanleitungen (einschließlich eventueller Software) darf unter den folgenden Bedingungen frei kopiert oder weitergegeben werden:

- es darf kein Preis für die Dokumentation (\*.zip) erhoben werden, außer einer angemessenen Kopiergebühr
- für einen kommerziellen Vertrieb der Baugruppen ist das Einverständnis des Autors erforderlich
- die Verwendung von Teilen der Dokumentation in eigenen Publikationen ist erlaubt, sofern ein eindeutiger Hinweis auf die Quelle erfolgt
- eine eventuell der Baugruppe beigelegte Software kann als Shareware konzipiert sein; dann ist sie zeitlich nicht limitiert, besitzt jedoch nur in der registrierten Version ihren vollen Funktionsumfang

Die Registrierung der Software ermöglicht Ihnen, Hinweise über die neueste (eventuell fehlerbehebene) Version zu erhalten. Wir haben uns bemüht, fehlerfrei zu programmieren. Die Erfahrung mit vielen Programmen zeigt aber, daß Software der „Version 1.0“ selten fehlerfrei funktioniert.



Hardwarefehler sind auf Grund der Anfertigung von Mustergeräten weitestgehend ausgeschlossen; Maßabweichungen bei der Bestückung können durch verschiedene Bauformen der Bauelemente auftreten.

Sollten Sie glauben, einen Fehler gefunden zu haben, senden Sie uns bitte per Email folgendes zu:

- eine kurze, aber präzise Beschreibung für die Reproduzierbarkeit des Fehlers
- eine Beschreibung der eingesetzten Hard- und Software

Ohne diese Informationen ist eine Bearbeitung nicht möglich.

Einige Programme laufen unter Visual Basic (VB) der Versionen 4/5. Die jeweilige \*.exe funktioniert daher nur unter Windows, wenn VB oder ein zusätzliches Softwarepaket mit den erforderlichen Dateien installiert ist.

Anwender, die darüber nicht verfügen, können sich eine Übersicht über die Funktionalität des Programms verschaffen. Das zusammengestellte Material beschreibt in Kurzform die Funktionalität der Software.

Da die entsprechenden Installationsroutinen für Anwender ohne VB recht groß sind (ca. 1 ... 5 MB), macht eine Verteilung über Online-Dienste kaum einen Sinn. In diesem Fall sollte per Email unter Angabe des Elektronikmoduls ein Informationsblatt angefordert werden.

In der Zukunft entstehende PC-Programme werden mit der Programmiersprache PROFAN entwickelt. Möglich sind auch Excel-Dateien mit Makrounterstützung.

Für Baugruppen, in denen Mikrocontroller (AVR der Fa. Microchip) eingesetzt werden, dient als SDK BASCOM (Basicdialekt).

BASCOM ist bis zu einer Programmgröße von 4 kByte ohne Einschränkungen verwendbar und unserer Meinung nach recht schnell zu beherrschen.

### **Ausnahme**

Die Software für die in dieser Dokumentation beschriebene Baugruppe SDM ist nicht in BASCOM geschrieben.

Hier kann ein normaler Editor zum Einsatz kommen. Anschließend muss das Programm mit einem C-Compiler für AVR kompiliert werden.

## 4.5 Technische Hinweise

### 4.5.1 Ausdrucken der Dateien

Der Ausdruck der Dateien sollte keine Probleme bereiten.

Im Gegensatz zu den Frontplattenbeschriftungen, die im Maßstab 1:1 vorliegen, sind die Leiterplattenzeichnungen der ersten Module im Maßstab 2:1 erstellt, was bei der Fertigung beachtet werden muß.

Für die Leiterplattenentwicklung jüngerer Datums wurde die vom Leiterplattenhersteller Beta-LAYOUT (<https://de.beta-layout.com/leiterplatten/technik/downloads/>) angebotene kostenfreie TARGET-Version eingesetzt.

Bei der Erstellung des Layouts der Leiterplatten wurde auf eine einfache Leitungsführung geachtet. Allerdings sind nicht mehr alle Bauelemente in einem DIL-Gehäuse verfügbar, so dass häufiger die SMD-Technik zum Einsatz kommt.

Die Bestückung gestaltet sich dann zwar etwas komplizierte, es kann aber Platz gespart werden, was geringere Kosten bei der Leiterplattenherstellung bedeutet.

Es ist bei der Leiterplattenbestellung ebenfalls möglich, einen Bestückungsauftrag zu erteilen.

### 4.5.2 Gehäuse und Frontplattenbeschriftungen

Sofern bei einer Baugruppe ein größeres Gehäuse verwendet wurde, handelt es sich um ein Kunststoff-Halbschalen-Gehäuse, was zum Beispiel über den CONRAD-Elektronikversand bezogen werden kann.

Diese Gehäuseform hat den Vorteil, daß mit geringem Aufwand eine ansprechende Frontplatte gestaltet werden kann. Die den jeweiligen Baugruppen beigefügten Datei „\* (Frontplatte).pdf“ beinhaltet jeweils einen Vorschlag. Sie können auf entsprechenden Papier (nicht zu dickes verwenden) ausgedruckt und unter Verwendung eines Klebestiftes auf die Frontplatte geklebt werden. Nach dem Trocknen kann noch eine selbstklebende Transparentfolie aufgeklebt werden. Mit einer Rasierklinge lassen sich jetzt überstehende Papier- und Folienreste schnell entfernen. Entsprechend den Konturen kann nun gekörnt und vorhandene Ausschnitte mit einer Laubsäge angefertigt werden. Nach Montage der Bedienelemente und der Verdrahtung mit der Leiterplatte kann die Montage des Gehäuses durchgeführt werden.

Stecker- oder Hutschienegehäuse können vom gleichen Anbieter, Reichelt-Elektronik oder direkt vom Hersteller (BOPLA) bezogen werden.

### 4.5.3 Bauelementeauswahl

Bauelemente ohne besonderen Hinweis sind als Vorschläge zu verstehen. An diesen Stellen können natürlich Typen verschiedener Hersteller zum Einsatz kommen, sofern sich ihre elektrischen Parameter gleichen.

Weitergehende Erläuterungen und Hinweise zum Aufbau elektronischer Schaltungen werden als bekannt vorausgesetzt bzw. können aus entsprechenden Literaturquellen bezogen werden.

### 4.5.4 SMD-Bestückung

Leider gibt es einige Bauelemente, die nicht in einem DIL-Gehäuse angeboten werden. Die Bestückung der Leiterplatte gestaltet sich daher komplizierter. Im Internet findet man zum Thema auch mehrere Anregungen und Tipps.

Zur Bestückung mit dem FTDI232RL wurde die LötKolbenspitze durch einen angespitzten 1,5 mm<sup>2</sup> Kupferdraht ersetzt. Die Leistung des LötKolbens mit 30 Watt ermöglicht ein gutes Fließen des Lötzinns. Eine Lupe, ausreichende Beleuchtung und eine ruhige Hand sind weitere Voraussetzungen für die Montage des ICs.

Anfragen, egal zu welcher Problematik, beantworten wir gerne per Email.

Viel Spaß beim Aufbau und Einsatz der Baugruppe wünscht Ihnen das

Amatronik Entwicklungsteam