

info@antrax.de - www.antrax.de



Iridium-Shield für Arduino und Clones

Arduino-Board Duemilanove (Atmega328) Arduino-Board UNO (Atmega328) Arduino-Board Mega2560 (Atmega2560) Clones

Bedienungsanleitung

15.01.2015

info@antrax.de - www.antrax.de



Inhaltsverzeichnis

Allgemeines	3
Lieferumfang	4
•Hardware	4
•Software	4
Systemvoraussetzungen	4
•Allgemein	4
•Programmierung	4
Technische Merkmale	5
•Allgemein	5
•Iridium	5
Iridium-Shield	6
Installation	7
•Hardware	7
•Software	7
Inbetriebnahme	8
•Bibliothek	8
Programmieren des Arduino-Boards	9
•Beispielprogramm "Iridium_Shield_Send"	11
•Beispielprogramm "Iridium_Shield_Receive"	12
Besonderheiten der Satellitenkommunikation	13
Links	14
Schaltpläne / Layouts	15

info@antrax.de - www.antrax.de



Allgemeines

Arduino ist eine Open-Source Physical-Computing-Plattform, die auf flexibler und leicht zu nutzender Hard- und Software basiert und zur Entwicklung von Prototypen dient. Das Arduino-Board besteht aus einem Mikrocontroller mit mehreren digitalen und analogen Ein- und Ausgängen und Schnittstellen. Die dazugehörige, auf einem PC laufende Entwicklungsumgebung wurde unter Java entwickelt und basiert auf Processing und anderer Open-Source-Software. Mit ihr können entwickelte Programme direkt auf das Arduino-Board hochgeladen werden.

Arduino hat eine riesige Fangemeinde und natürlich eine eigene Homepage mit vielen Beispielapplikationen und umfangreichem Forum \rightarrow <u>http://www.arduino.cc/</u>

Um die Funktionalität des *Arduino*-Boards zu erweitern und um andere Geräte zu steuern, können sog. Shields (Erweiterungsbaugruppen) auf das *Arduino*-Board gesteckt werden. Mit dem *Iridium-Shield* wird das *Arduino*-Board um die Funktionalität "echter Satellitenfunk über das Iridium-Netzwerk" erweitert. Das Iridium-System bietet eine weltweit einzigartige Netzverfügbarkeit.

Das *Iridium-Shield* wird auf das *Arduino*-Mainboard aufgesteckt und kommuniziert mit dem Mikrocontroller über die serielle Schnittstelle. Als einzige externe Komponente ist eine Iridiumantenne erforderlich (über einen SMA-Steckverbinder).

Bitte lesen Sie diese Beschreibung vor der ersten Inbetriebnahme komplett und sorgfältig. Sie beschreibt den bestimmungsgemäßen Gebrauch und enthält wichtige Hinweise zur Installation/Inbetriebnahme des *Iridium-Shields* mit dem *Arduino-Board*. Für die Folgen des nicht bestimmungsgemäßen Gebrauchs übernimmt der Hersteller keine Haftung. Sämtliche Garantieansprüche entfallen.

info@antrax.de - www.antrax.de

X

Lieferumfang

Hardware

Alles komplett? Die Einzellieferung eines *Iridium-Shields* beinhaltet:

- das Iridium-Shield
- diese Bedienungsanleitung
- ein Iridium SBD-Modul vom Typ "Iridium 9602 SBD Transceiver" (optional)
- das Arduino-Board (optional)
- eine Iridium-Antenne (optional)
- ein Steckernetzteil (optional)

Software

Software-Pack (als Download) mit folgendem Inhalt:

- Eine Bibliothek zur Verwendung des Iridium-Shields
- Beispielprogramm für die Nutzung des Iridium-Moduls
- Arduino-Board / Iridium-Shield Design und Schaltplan

HINWEIS: Die URL zu der *Arduino*-Entwicklungsumgebung (inkl. Treiber) ist unter <u>Links</u> verfügbar.

Systemvoraussetzungen

Allgemein

- Arduino-Board "Duemilanove", "UNO" oder "Mega2560" oder geeignete Clones
- Iridium-Vertrag bei einem Airtime-Provider (wenn gewünscht unterstützen wir Sie bei der Beurteilung von Iridium-Verträgen verschiedener Airtime-Provider)

Programmierung

- Arduino-Entwicklungsumgebung
- USB-Schnittstelle
- Windows / Mac OS X / Linux: 32 oder 64 bit



Technische Merkmale

Allgemein

- Betriebsspannung: 5VDC ... 9VDC
- erforderliche Mainboards: Duemilanove/Uno, Mega2560 oder Arduino-Clones

Iridium

- Steckplatz für ein Iridium 9602 SBD Transceiver
 - o verwendbar mit Vertrag eines Airtime-Providers
 - Ansteuerung über die serielle Schnittstelle (TTL-Pegel)
- genaue Beschreibung des Iridium-Transceivers siehe hier ---> http://www.antrax.de/downloads/iridium-shield/quake-datasheets/

info@antrax.de - www.antrax.de



Iridium-Shield



info@antrax.de - www.antrax.de

X

Installation

Hardware

- Iridium Modul auf das Iridium-Shield stecken
- Iridiumantenne anschließen
- Iridium-Shield auf das Arduino-Board stecken.
- Netzteil an das *Arduino*-Board anschließen und *Arduino*-Board via USB-Kabel mit einem PC verbinden

ACHTUNG:

Es ist wichtig, mit der Iridium-Antenne eine gute Sicht zum Himmel zu haben (die Beschriftung "SKY SIDE" auf den Antennengehäusen ist ernstzunehmen!). Ein schlechte Satellitenverbindung ist fast immer auf eine ungünstige Antennenplazierung oder -anschaltung zurückzuführen. Setzen Sie nur geeignete Verlängerungskabel ein, z.B. Aircom Plus oder LMR®-400-Kabel und hochwertige Steckverbinder von Markenherstellern wie Huber+Suhner, Radiall oder Telegärtner. Einfache RG-58- oder RG-174-Verlängerungskabel sind meist <u>nicht</u> geeignet!

Iridium-Antennen und deren Anschaltung sind deutlich anspruchsvoller als z.B. Mobilfunk-Antennen.

Software

Treiber zu dem *Arduino*-Board aus dem Verzeichnis der *Arduino*-Entwicklungsumgebung installieren (falls nötig).

info@antrax.de - www.antrax.de

X

Inbetriebnahme

Bibliothek

Um das Arbeiten mit dem *Iridium-Shield* so einfach wie möglich zu gestalten, wurde eine Bibliothek zu diesem Shield entwickelt.

Diese Bibliothek und zwei Beispielprogramme für des Senden und Empfangen von Nachrichten finden Sie unter dem Punkt "Downloads" auf der Artikelseite des <u>Iridium-Shields</u> in unserem Shop. Nach dem Entpacken der Datei kopieren Sie den Inhalt bitte in das Verzeichnis "libraries" des Arduino-Pfades (z.B. "f:\arduino-1.0.1\libraries\").

Lokaler Datenträger (C:) Programme arduino-0022 libraries							
n Neuer Ordner							
Januaria ArduinoTestSuite							
EEPROM							
📔 Ethernet							
冯 Firmata							
🍑 GSM_GPS_Shield_Mega							
GSM_GPS_Shield_UNO_DUEMILANOVE							
길 Iridium_Shield_Mega							
Iridium_Shield_UNO_DUEMILANOVE							
📙 LiquidCrystal							
🌗 Matrix							
OBD2_Shield							
) SD							
鷆 Servo							
SoftwareSerial							
SDI SDI							

Die Bibliothek kann mit 3 Klicks dem aktuell geöffneten Programm hinzugefügt werden (#include <Iridium.h>). Natürlich kann diese Bibliothek für die eigenen Bedürfnisse geändert oder erweitert werden.

Iridium_Shield Arduino 0022							
Sketch Tools Help							
Verify / Compile Strg+R							
Stop							
Show Sketch Folder Strg+K							
Import Library 🕨	ArduinoTestSuite						
Add File	EEPROM						
: antrax Datentechnik G	Ethernet						
th: Arduino UNO (ATmega32)	Firmata						
	GSM_GPS_Shield_Mega						
a attack block be	GSM_GPS_Shield_UNO_DUEMILANOVE						
e <liidium.n></liidium.n>	Iridium_Shield_Mega						
(14, 19200);	Iridium_Shield_UNO_DUEMILANOVE						



Programmieren des Arduino-Boards

Das **Iridium-Shield** wird immer von der auf dem *Arduino*-Board laufenden Software angesprochen bzw. gesteuert. Das Übertragen der Programme auf das *Arduino*-Board ist mit der *Arduino*-Entwicklungsumgebung ganz einfach. Im folgenden werden kurz die dafür benötigten Schritte beschrieben:

• das mit dem *Iridium-Shield* bestückte *Arduino*-Mainboard über die USB-Schnittstelle mit dem PC verbinden (die USB-Treiber sollten bereits installiert worden sein)

+71 - 1 A - 4

- Öffnen der Arduino-Entwicklungsumgebung
 - unter File \rightarrow Examples
 - $\circ \quad \text{Iridium}$
 - Beispielprogramm
- da das Arduino-Board (trotz USB-Anschluss) im PC als serielles Gerät angezeigt wird, muss unter
 - $\circ \quad \text{Tools} \rightarrow \text{Serial Port}$

die verwendete Schnittstelle gewählt werden

Skelen_octoria Alidanio 0022									
e	Edit Sketch Tools H	lelp							
	New	Strg+N							
	Open	Strg+O							
	Sketchbook								
	Examples	Þ		1.Basics	•				
	Close	Strg+W		2.Digital	•				
	Save	Strg+S		3.Analog	•				
	Save As	Strg+Umschalt+S		4.Communication	•				
	Upload to I/O Board	Strg+U		5.Control	•				
	Dage Setup	Stra+Upschalt+P		6.Sensors	•				
	Print	Stra+P		7.Display	•				
		Sugar		8.Strings	•				
	Preferences	Strg+Comma		ArduinoISP					
	Quit	Strg+Q		GSM_and_OBD	۲				
				ArduinoTestSuite	•				
				EEPROM	•				
				Ethernet	•				
				Firmata	•				
				GSM_GPS_Shield_Mega	+				

GSM_GPS_Shield_Mega GSM_GPS_Shield_UNO_DUEMILANOVE Iridium_Shield_Mega Iridium_Shield_UNO_DUEMILANOVE Using_Iridium_Shield



- über den Upload-Button (im Bild weiß hinterlegt) kann das geänderte Programm direkt auf das Arduino-Board hochgeladen werden
- nach dem im unteren Teil des Fensters die Meldung "Done." erscheint, ist die Übertragung erfolgreich abgeschlossen worden



- das Programm wird automatisch gestartet
- über die serielle Schnittstelle in der Arduino-Entwicklungsumgebung kann die Kommunikation zwischen dem Iridium Modul und dem Arduino Board beobachtet und ggf. empfangene Nachrichten angezeigt werden. (ACHTUNG: Beim öffnen des Monitors startet das Programm von vorne!)

Hinweis: Das **Iridium-Shield** muss immer mit einem geeigneten externen Netzteil betrieben werden. Die Stromaufnahme des Iridium-Moduls beträgt bis zu 1,5A. Die Stromversorgung über das USB-Kabel ist dazu nicht ausreichend!



Beispielprogramm "Iridium_Shield_Send"

Das Beispielprogramm "Iridium_Shield_Send" zeigt, wie eine SBD-Message vom *Iridium-Shield* versendet wird.

Die Demosoftware macht der Reihe nach folgendes:

- Initialisierung des Arduino-Systems
- Einschalten des *Iridium-Shields* über Pin 14 (Pin 54 beim Arduino Mega 2560) des Arduino-Boards. Die LED2 leuchtet.
- zu übertragenden Text in den Sendebuffer schreiben (Mobile Originated Buffer MOB). Im MOB kann immer nur eine Nachricht gespeichert werden, d.h. eine ggf. noch im Sendespeicher befindliche Nachricht wird mit der neuen Nachricht überschrieben.
- Sendevorgang anstoßen (beinhaltet den Satelliten-Connect und den eigentlichen Sendevorgang). Dieser Vorgang dauert, je nach Empfangslage, zwischen einigen Sekunden und etlichen Minuten. Der Datenaustausch zwischen *Iridium-Shield* und Satellit wird SBD-Session genannt. Dabei wird die im Sendespeicher befindliche Nachricht an den Satelliten gesendet (Achtung: die Nachricht selbst bleibt im Sendespeicher erhalten, sie wird nur kopiert und versendet).

Ferner wird während der SBD-Session geprüft, ob eine Nachricht auf dem Satelliten für das Iridium-Modul vorliegt. Ist dies der Fall, wird diese vom Satelliten in den Empfangsspeicher (Mobile Terminated Buffer – MTB) des Moduls transferiert. Im Empfangsspeicher des Moduls kann ebenfalls nur eine Nachricht gespeichert werden, womit eine dort vorhandene Nachricht immer überschrieben wird.

- Ausgabe der Ergebnismeldung (gesendet/nicht gesendet)
- Abschalten des *Iridium-Shields*

Die LED1 signalisiert, ob das Iridium-Modul Funkempfang hat (dafür benötigt das Modul nach Power-On möglicherweise einige Minuten).

Die gesamte Kommunikation mit dem Iridium-Modul erfolgt über einfache AT-Befehle im Klartext. Für eine ausgefeiltere Kommunikation können die in den Libraries aufgeführten Demo-Routinen einfach modifiziert werden.

info@antrax.de - www.antrax.de

Die Demoroutinen kommunizieren mit dem Iridium-Modul folgendermaßen (Trace auf der serielle Schnittstelle):

(blau = Arduino-Mainboard ---> Iridium-Modul)
(rot = Iridium-Modul ---> Arduino-Mainboard)T





info@antrax.de - www.antrax.de

OK AT+SBDI +SBDI: 1, 337, 0, 0, 0, 0 OK AT+SBDD0 0 OK Mo buffer cleared! Your Message sent successfully!



- 13 -





Beispielprogramm "Iridium_Shield_Receive"

Im Beispielprogramm "Iridium_Shield_Receive" wird gezeigt, wie eine SBD-Message vom **Iridium-Shield** empfangen wird.

Die Demosoftware macht der Reihe nach folgendes:

- Initialisierung des Arduino-Systems
- Einschalten des *Iridium-Shields* über Pin 14 (Pin 54 beim Arduino Mega 2560) des Arduino Boards. Die LED2 leuchtet.
- Empfangsvorgang anstoßen (beinhaltet den Satelliten-Connect, eine Stausabfrage und den eigentlichen Empfangsvorgang). Dieser Vorgang dauert, je nach Empfangslage, zwischen einigen Sekunden und etlichen Minuten. Der Datenaustausch zwischen Iridium Modul und Satellit wird SBD-Session genannt. Dabei fragt das Iridium-Modul den Satelliten, ob eine Empfangsmessage für die eigene Modul-ID vorliegt.

Ist dies der Fall, wird diese vom Satelliten in den Empfangsspeicher (Mobile Terminated Buffer – MTB) des Moduls transferiert. Im Empfangsspeicher des Moduls kann nur eine Nachricht gespeichert werden, womit eine dort möglicherweise noch vorhandene Nachricht immer überschrieben wird.

- Ausgabe der Ergebnismeldung und des empfangenen Textes
- Abschalten des *Iridium-Shields*

Die LED1 signalisiert ob das Iridium Modul Empfang hat (dafür benötigt das Modul nach Power-On möglicherweise einige Minuten).

Die zu übertragende Message muss vorher als Anhang (als Textdatei) einer Email an die Adresse "<u>data@sbd.iridium.com</u>" gesendet worden sein. Im Betreff der Email muss die ID des Iridium-Moduls angegeben werden, an das die Message geschickt werden soll.

info@antrax.de - www.antrax.de

Die Demoroutinen kommunizieren mit dem Iridium-Modul folgendermaßen (Trace auf der serielle Schnittstelle):

(blau = Arduino-Mainboard ---> Iridium-Modul)
(rot = Iridium-Modul ---> Arduino-Mainboard)T





info@antrax.de - www.antrax.de

+SBDRT: Hello IRIDIUM-Modul! OK AT+SBDD1 0 OK Mt buffer cleared! Received Message: **Hello IRIDIUM-Modul!** OK





Besonderheiten der Satellitenkommunikation

Satellitenkommunikation ist teilweise mit der Kommunikation im Bereich Mobilfunk vergleichbar, weist aber auch eklatante Unterschiede auf:

- Das Abrechnungssystem der Airtime-Provider ist teilweise anders als bei Mobilfunkprovidern. So muss z.B. die Abfrage, ob eine SBD-Message vorliegt, aktiv erfolgen. Außerdem ist diese Abfrage bereits kostenpflichtig. Sie sollte deshalb möglichst mit dem Versenden von SBD-Messages kombiniert werden
- Es gibt "Sperrfristen" (3 Minuten), die das zu häufige Einbuchen in das Satellitennetz verhindern sollen
- das Iridium-Modul kann eine SBD-Message immer nur an eine (einzige) Email-Adresse versenden. Diese Email-Adresse kann/muss bei der Einrichtung des Anschlusses angegeben werden

info@antrax.de - www.antrax.de



Links

Weitere Informationen finden Sie

- auf unserer Internetseite \rightarrow <u>http://www.antrax.de</u>
- Arduino-Entwicklungsumgebung → <u>http://arduino.cc/en/Main/Software</u>
- Schaltplan und Layout → <u>http://www.antrax.de/downloads/iridium-shield/antrax-hardware/</u>
- Iridium Library → <u>http://www.antrax.de/downloads/iridium-shield/antrax-examples/</u>
- genaue Beschreibung des Iridium-Transceivers und die SBD-Beschreibung siehe hier → http://www.antrax.de/downloads/iridium-shield/quake-datasheets/