

Protokolle: ISO9141-2 und ISO14230-2 (KWP2000)

Automatische Protokollerkennung

Slow-Init (5 Baud) und Fast-Init

Automatischer oder manueller Init

Serielle Schnittstelle mit 9600 oder 38400 Baud, 8N1, kompatibel zu MAX232 und USB/RS-232-Wandler-Bausteinen

Quarz-Oszillator 7,3728MHz

Schneller AVR-Controller von ATMEL®

AT-Befehlssatz, hierdurch kompatibel zu vielen existierenden Programmen

Multiframe kompatibel

Nur wenige externe Bauteile erforderlich

K/KL-Transit-Modus per Taster aktivierbar

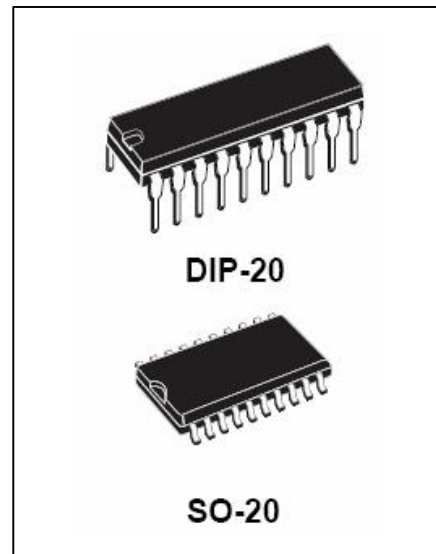
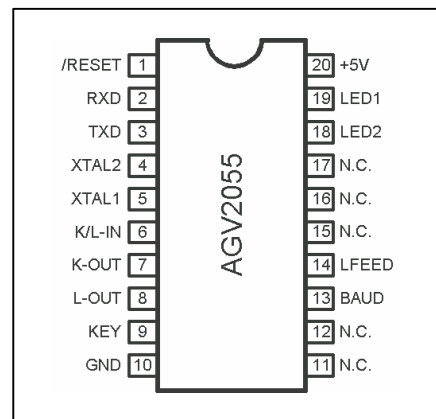
2 Leuchtdioden für Connect und Datenfluß anschließbar

Jeder ausgelieferte Chip besitzt eine einzigartige unveränderliche Seriennummer

Veränderbarer Identifikationstext zur Personalisierung des Chips

Lieferbar im DIL20- oder SO20-Gehäuse

Made in Germany



Beschreibung

Der AGV2055 ist ein OBD2 nach RS232-Interpreter-Baustein modernster Bauart. Er basiert auf dem schnellen AVR-Controller ATTiny2313 von ATMEL®. Mit nur wenigen externen Bauteilen kann einfach ein funktionsfähiges OBD2-Interface aufgebaut werden. Der AGV2055 Controller unterstützt die gängigen Protokolle ISO9141-2 und KWP2000 für viele europäische Fahrzeuge. Zusätzlich ist ein K/KL-Transit-Modus im Bios des Controllers enthalten, der bei vielen Fahrzeugen des VAG-Konzerns mit Hilfe der Software VAG.com bis Version 311 benutzt werden kann.

Mit seinem AT-Befehlssatz kann der Controller vom PC mit einem Terminalprogramm (wie z.B. Hyperterm) angesprochen werden, es ist keine spezielle Software notwendig. Aufgrund der Befehls-Kompatibilität zu anderen existierenden Controllern können eine ganze Reihe bereits existierender Programme auf dem PC verwendet werden.

Die PC-Schnittstelle kann mit einem handelsüblichen MAX232 als RS-232-Pegelwandler oder mit einem USB nach RS232-Wandler (z.B. FT232BM von FTDI) bestückt werden. Beide Schaltungsvarianten finden Sie in diesem Datenblatt.

Pinbelegung des Controllers

Reset (Pin 1)

Zurücksetzen des Controllers durch einen negativen Impuls von mindestens 2,5µs. Dieser Pin kann in der Regel unbeschaltet bleiben, da der Controller eine interne Reset-Logik besitzt.

RXD (Pin 2)

Serieller Eingang (RS-232). Protokoll 9600,8N1. Invertiertes Signal, kann direkt mit dem Ausgang eines MAX232 oder FT232BM verbunden werden. Bitte diesen Pin nicht direkt mit der seriellen Schnittstelle des PC verbinden, da die Spannungspegel den Chip zerstören würden.

TXD (Pin 3)

Serieller Ausgang (RS-232). Protokoll 9600,8N1. Invertiertes Signal, kann direkt mit dem Eingang eines MAX232 oder FT232BM verbunden werden.

XTAL2, XTAL1 (Pin4, 5)

Anschluß für einen Quarz 7,3728 MHz. Vom Chiphersteller wird empfohlen, Kondensatoren 22...27pF an beiden Pins gegen Masse zu legen.

K/L-IN (Pin 6)

Eingangssignal für die ODER-Verknüpften K- und L-Leitungen. Signal muß invertiert werden.

K-OUT (Pin 7)

Ausgangssignal für die K-Leitung. Zur Ansteuerung eines Kleinleistungstransistors BC547 (mit Vorwiderstand).

L-OUT (Pin 8)

Ausgangssignal für die L-Leitung. Zur Ansteuerung eines Kleinleistungstransistors BC547 (mit Vorwiderstand).

KEY (Pin 9)

Anschluß für einen Taster zur Umschaltung des Controllers in den K/KL-Transit-Modus. Dieser Pin muß während der Reset-Phase (also bei Einstecken des Interface auf den OBD-Anschluß) auf Masse gelegt werden, damit der Controller sicher in den K/KL-Modus geschaltet werden kann. Die ständig leuchtende LED1 (grün) zeigt den aktivierten Modus an.

GND (Pin 10)

Massesignal.

N.C. (Pin 11)

Pin wird nicht benutzt. Bitte nicht beschalten, da dieser Pin eventuell bei einer späteren Version benutzt werden kann.

N.C. (Pin 12)

Pin wird nicht benutzt. Bitte nicht beschalten, da dieser Pin eventuell bei einer späteren Version benutzt werden kann.

BAUD (Pin 13) * **AB BIOS 4.0** *

Baudrate der seriellen Schnittstelle wählen.
Pin offen oder auf +5V = 9600 Baud
Pin auf Massepotential (GND) = 38400 Baud

LFEEED (Pin 14)

LineFeed-Modus der seriellen Schnittstelle. Mit offenem Pin wird als Zeilenendekennung über die serielle Schnittstelle CR/LF (13,10) ausgegeben. Wird der Pin auf Masse gelegt, wird als Zeilenendekennung nur CR (13) ausgegeben. Dieser Pin wird nur beim Reset abgefragt. Der LineFeed-Modus kann softwaremäßig nachträglich mit dem Befehl ATL0 bzw. ATL1 geändert werden.

N.C. (Pin 15)

Pin wird nicht benutzt. Bitte nicht beschalten, da dieser Pin eventuell bei einer späteren Version benutzt werden kann.

N.C. (Pin 16)

Pin wird nicht benutzt. Bitte nicht beschalten, da dieser Pin eventuell bei einer späteren Version benutzt werden kann.

N.C. (Pin 17)

Pin wird nicht benutzt. Bitte nicht beschalten, da dieser Pin eventuell bei einer späteren Version benutzt werden kann.

LED2 (Pin 18)

Anschluß einer Leuchtdiode (Rot, über Vorwiderstand). Leuchtet bei erfolgreichem Connect.

LED1 (Pin 19)

Anschluß einer Leuchtdiode (Grün, über Vorwiderstand). Blinkt bei Datentransfer von/zur OBD2-Schnittstelle.

+5V (Pin 20)

Anschluß für die 5 Volt Stromversorgung. Stromaufnahme des Controllers max. 10mA.

Kommunikation mit dem Controller

Die Kommunikation des PC mit den AGV-Controllern geschieht über die serielle Schnittstelle. Der USB-Anschluß des AGV-Interface wird als virtuelle serielle Schnittstelle auf dem PC eingerichtet, so dass Standardsoftware auf einen COM-Port zugreifen kann. Die Datenrate ist abhängig von der Belegung des Pin 13 (BAUD) auf 9600 oder 38400 Baud, 8 Datenbits, 1 Stopbit, kein Handshake (9600/38400, 8N1) eingestellt.

Wenn Sie ein Terminalprogramm auf dem PC starten, die richtige Baudrate und den COM-Port wählen, sollte nach Einstecken des Interface in die OBD2-Buchse des Fahrzeugs auf dem PC folgender Text erscheinen:

```
ELMsfire AGV2055 v4.0
```

```
>
```

Hinweis: Diese Meldung variiert je nach Controller-Typ und kann zusätzlich mit dem Tool „AGV-Internals“ geändert werden.

Welche Meldung auch erscheint, sie zeigt an, daß die Kommunikation des Interface mit dem PC funktioniert. Das Zeichen „>“ bedeutet, daß das Interface bereit ist, Befehle zu empfangen. Das Interface unterscheidet nun zwei verschiedene Kommandogruppen:

1. Interne Befehle zur Konfiguration und Initialisierung des Interface-Controllers. Alle diese Befehle beginnen mit den Zeichen „AT“, dieses wurde von den Steuerbefehlen bei Modems übernommen und bedeutet „Attention, Achtung“.
2. Daten, die an den OBD2-Bus für das Steuergerät des Fahrzeugs weitergeleitet werden. Alle diese Befehle werden als hexadezimale Zahlen übermittelt, es dürfen deshalb nur ASCII-Zeichen 0-9 und A-F paarweise eingegeben werden.

Alle eingegebenen Befehle müssen mit einem Zeilenende-Zeichen (Carriage Return, Dez. 13, Hex \$0D) abgeschlossen werden. Leerzeichen oder Tabulatoren werden automatisch herausgefiltert, Klein- und Großschreibung wird nicht unterschieden. In den folgenden Beispielen muß jede

Eingabe mit dem Zeilenende-Zeichen abgeschlossen werden, es wird nicht extra angegeben.

Beispiele:

```
at si  
Wird intern nach ATSI gewandelt
```

```
A T Z  
Wird intern nach ATZ gewandelt
```

```
01 1c  
Wird intern nach 011C gewandelt
```

Falls das Zeilenende-Zeichen ausbleibt, wird der Befehl automatisch nach 5 Sekunden abgebrochen und ein „?“ wird als Fehlermeldung ausgegeben.

Befehle, die der Controller nicht versteht, werden ebenfalls mit einem „?“ als Fehlermeldung quittiert.

Einige interne AT-Befehle sowie die Dateneingabe für den OBD2-Bus werden mit Werten übermittelt, die aus 2-stelligen HEX-Zeichen zusammengesetzt sind. Auch wenn der zu übermittelnde Wert nur einstellig ist muß er immer 2-stellig mit führender Null eingegeben werden.

Beispiele:

```
ATST 80  
Richtig! Der Wert $80 wird übermittelt
```

```
ATSW8  
Falsch! Es muß immer eine grade Anzahl an Hexzeichen eingegeben werden.
```

Obwohl die AGV-Controller weitgehend denselben Befehlssatz wie die ELM-Controller benutzen (damit ist gewährleistet, daß Standardprogramme funktionieren) gibt es doch einige Unterschiede. Es fehlen bei den AGV-Controllern einige Befehle, andere sind zusätzlich vorhanden. Einige fehlende Befehle konnten nicht integriert werden, da sie im Multiprotokoll-Betrieb nicht möglich gewesen wären. Hinzu kamen dafür spezielle Befehle für den Multiprotokoll-Betrieb sowie einige Konfigurationsbefehle für die besonderen Features der AGV-Controller, wie die Änderung des Identifikationstextes und das Auslesen und Setzen spezieller Parameter aus dem bzw. in den Controller. Alle Befehle wurden jedoch soweit kompatibel gehalten, damit Standardsoftware einwandfrei mit den AGV-Controllern funktioniert.

Der AT-Befehlssatz des Controllers**>ATZ**

Dieser Befehl bewirkt einen sofortigen Abbruch aller laufenden Funktionen und einen Warmstart des Controllers. Alle Parameter werden in den Grundzustand gesetzt und der Selbsttest wird durchgeführt. Zum Schluß wird der Ident-Text ausgegeben.

Ausgabe:

ELMsfire AGV2055 v4.0

>

Der Identifikationstext kann mit dem Tool AGV-Internals geändert werden.

>ATI

Hiermit wird nur der Identifikationstext ausgegeben, ohne einen Warmstart durchzuführen. Laufende Funktionen, wie das automatische Wakeup bei ISO oder KWP2000, werden nicht abgebrochen.

Ausgabe:

ELMsfire AGV2055 v4.0

>

Der Identifikationstext kann mit dem Tool AGV-Internals geändert werden.

>ATD

Alle Parameter werden in den Grundzustand wie nach einem Kalt- oder Warmstart versetzt. Dies betrifft die Parameter Echo (ATE), Linefeed (ATL), Header (ATH), Wakeup-Zeit (ATSW), Empfangs-Adresse (ATSR), KWP-Headerbytes (ATSH).

Ausgabe:

OK

>

>ATE0**>ATE1**

Dieser Befehl schaltet das serielle Echo ein (1) oder aus (0). Alle Daten, die über die serielle Schnittstelle empfangen werden, werden bei eingeschaltetem Echo wieder zum PC geschickt.

Ausgabe:

OK

>

>ATL0**>ATL1**

Über diesen Befehl wird die Übermittlung des Linefeed-Zeichens am Ende der Zeile ein- (1) oder ausgeschaltet (0). Alle zum PC gesendeten Antworten sind in der Regel nur mit einem Carriage-Return abgeschlossen (13 Dez., \$0D Hex). Mit eingeschaltetem Linefeed wird hinter jedem Carriage-Return noch das Linefeed-Zeichen (10 Dez., \$0A Hex) ausgegeben. Besonders wenn man mit einem Terminalprogramm die Befehl von Hand übermittelt, ist es recht sinnvoll, den Linefeed eingeschaltet zu lassen, da sonst alle Antworten in einer Zeile dargestellt werden und neue Antworten die Alten überschreiben.

Ob der Linefeed nach einem Reset, dem Kommando ATZ oder ATD ein- oder ausgeschaltet ist, kann durch Belegung des PIN14 (LFEED) voreingestellt werden. Ist dieser Pin offen oder mit +5V verbunden, hat dies dieselbe Bedeutung wie ATL1, wird der Pin auf Massepotential (GND) gelegt, ist ATL0 voreingestellt.

Ausgabe:

OK

>

>ATH0**>ATH1**

Mit diesem Befehl kann eingestellt werden, ob bei OBD2-Antworten der Header und das Checksummenbyte mit ausgegeben werden soll.

Ausgabe:

Mit ATH0:

>0100

41 00 E8 19 30 12

Mit ATH1:

>0100

48 6B 10 41 00 E8 19 30 12 47

48, 68, 10 sind die 3 Headerbytes
47 ist das Checksummenbyte

Ausgabe:

OK

>

>ATPD

Nicht mehr in Bios-Version ab 4.0

>ATFD

Nicht mehr in Bios-Version ab 4.0

>ATBD

Die Abkürzung von „**B**uffer **D**ump“ bewirkt, den Inhalt des internen OBD2-Sende- und Empfangsspeicher auszugeben. Gültige Daten stehen hier jedoch nur, wenn zuvor ein OBD2-Befehl ausgeführt wurde.

Beispiel:

```
>0100  
41 00 E8 19 30 12
```

>ATBD

```
0A 48 6B 10 41 00 E8 19 30 12 47 00
```

Das erste Byte gibt die Anzahl der gültigen Zeichen im Speicher an. In diesem Fall handelt es sich um 10 Zeichen (Hex 0A). Das letzte Byte ist in diesem Fall ungültig und kann einen beliebigen Wert beinhalten.

>ATSI

Durch Eingabe dieses Befehls wird der AGV-Controller versuchen, einen SLOW-Init zu einem ISO9141 oder KWP2000 Fahrzeug auszuführen. Dies gelingt jedoch nur, wenn der Fahrzeugbus eines dieser Protokolle beherrscht und einen SLOW-Init erwartet. Die Ausführung des SLOW-Init benötigt ca. 2,5 Sekunden.

Beispiel:

```
>ATSI  
BUS INIT: ...OK
```

Sollte das Fahrzeug keinen SLOW-Init unterstützen, erscheint folgende Fehlermeldung:

Beispiel:

```
>ATSI  
BUS INIT: ...ERROR
```

Nach einem Disconnect mit ATZ kann es bis zu 5 Sekunden dauern, bis ein erneuter SLOW-Init möglich ist. Man sollte auf jeden Fall diese Zeit abwarten, bevor ein neuer Init-Versuch gestartet wird.

Dieser Befehl ist nur zu Experimentierzwecken vorhanden. Der AGV-Controller führt einen automatischen Init und damit eine automatische Ermittlung des unterstützten Protokolls durch, sobald ein OBD2-Befehl eingegeben wird.

>ATFI

Durch Eingabe dieses Befehls wird der AGV-Controller versuchen, einen FAST-Init zu einem Fahrzeug mit KWP2000-Protokoll auszuführen. Dies gelingt jedoch nur, wenn der Fahrzeugbus dieses Protokoll beherrscht und einen FAST-Init erwartet.

Beispiel:

```
>ATFI  
BUS INIT: OK
```

Sollte das Fahrzeug keinen FAST-Init unterstützen, erscheint folgende Fehlermeldung:

Beispiel:

```
>ATFI  
BUS INIT: ERROR
```

Nach einem Disconnect mit ATZ kann es bis zu 5 Sekunden dauern, bis ein erneuter FAST-Init möglich ist. Man sollte auf jeden Fall diese Zeit abwarten, bevor ein neuer Init-Versuch gestartet wird.

Dieser Befehl ist nur zu Experimentierzwecken vorhanden. Der AGV-Controller führt einen automatischen Init und damit eine automatische Ermittlung des unterstützten Protokolls durch, sobald ein OBD2-Befehl eingegeben wird.

>ATST xx

Nicht mehr in Bios-Version ab 4.0

Alle Zeitparameter sind kompatibel zu den entsprechenden SAE-Normen und können nicht verändert werden.

>ATSA xx

Neu ab Bios-Version 4.0

Einstellung der Empfangsadresse für OBD-Antworten. Vor allem bei Fahrzeugen mit KWP2000 Protokoll kann es passieren, daß mehrere ECUs (Steuergeräte) auf einen „Functional Request“ antworten. Mit diesem Befehl ist es möglich, nur die Antworten des gewünschten ECUs auszufiltern.

Beispiel:

KWP2000 mit eingeschalteten Header ATH1

```
>0100
C6 F1 10 41 00 11 22 33 44 CS
C6 F1 18 41 00 11 22 33 44 CS
```

Hier haben auf den „Functional Request“ 2 ECUs geantwortet. Das 3. Byte jeder Zeile gibt die ECU-Adresse an. Um bei zukünftigen Antworten nur noch diejenigen des 1. ECU auszufiltern, muß folgender Befehl eingegeben werden:

```
>ATSA 10
```

Wird hier ein falscher Wert eingegeben, der zu keiner ECU-Adresse paßt, wird nur noch „NO DATA“ angezeigt.

Ein Ausschalten der Filteradresse ist mit ATSA 00, ATD, ATZ möglich.

>ATSW xx

Mit diesem Befehl kann die Zeitdauer zwischen den automatischen Wakeup-Befehlen bei einer bestehenden ISO9141 oder KWP2000 Verbindung eingestellt werden. ISO9141 und KWP2000 Fahrzeugcontroller erwarten einen regelmäßigen Datenverkehr zum angeschlossenen OBD2-Interface. Sollte dieser längere Zeit ausbleiben (laut SAE-Norm 5000ms), wird die Verbindung getrennt und muß durch einen erneuten SLOW- oder FAST-Init wieder hergestellt werden.

Dieser Befehl erwartet die Zeitdauer als 2-stelligen HEX-Code. Die Zeitdauer zwischen den Wakeup-Commandos ergibt sich aus dem Hex-Wert * 20 Millisekunden.

Beispiel:

```
>ATSW60
OK
```

>

Hier wird eine Wakeup-Zeit von \$60 (96) * 20ms = 1,92 Sekunden eingestellt. Der Standardwert wird mit ATZ und ATD auf \$7D (125) eingestellt, was genau 2,5 Sekunden ent-

spricht. Wird als Wert 0 eingetragen, ist der automatische Wakeup abgeschaltet und die Verbindung wird nach 5 Sekunden getrennt, wenn keine OBD2-Kommandos übertragen werden.

>ATSH xx yy zz

Neu ab Bios-Version 4.0

Zu Experimentierzwecken können die gesendeten OBD2-Headerbytes verändert werden.

Die AGV-Controller erzeugen standarmäßig folgende Headerbytes:

```
ISO9141: 68 6A F1
KWP2000: Cn 33 F1
          (n=Länge, wird automatisch vom Interface eingesetzt)
```

Beispiel:

```
>ATSH 80 10 F1
OK
```

>

Hier wird der Header auf „Physical Request“ an das Steuergerät mit der Adresse 10 eingestellt.

Bei KWP2000 können die Headerbytes schon vor der Initialisierung mit ATFI oder der automatischen Protokollsuche geändert werden, bei ISO9141 ist dies erst nach erfolgreichem Init möglich.

Ein zurücksetzen der Headerbytes auf die Standardwerte ist nur mit ATZ möglich.

>ATC

Geändert ab Bios-Version 4.0

Anzeige des aktuell benutzten Protokolls als Hexwert.

Ab Bios Version 4.0 wird hier ein zu vorherigen Versionen unterschiedlicher Wert angezeigt. Zur Unterscheidung ist das obere Nibble des Wertes immer \$Fx, das untere Nibble gibt das aktuelle Protokoll an.

Beispiel:

```
>ATC
F3
```

>

```
$F0: Kein Protokoll aktiv (z.B. nach ATZ)
$F3: ISO9141-2 (5 Baud Init)
$F4: KWP2000 (5 Baud Init)
$F5: KWP2000 (Fast Init)
```

>ATDP

Anzeige des aktuell benutzten Protokolls im Klartext.

Beispiel:

```
>ATDP  
ISO 9141-2 (5 Baud)
```

>

Hier die Liste alle möglichen Ausgaben:

```
ISO9141-2 (5 Baud Init)  
KWP2000 (5 Baud Init)  
KWP2000 (Fast Init)
```

Ist kein Protokoll aktiv (z.B. nach ATZ) wird folgender Text ausgegeben:

```
NOT CONNECTED
```

>AT!00

Ausgabe der Seriennummer des AGV-Controllers. Alle AGV-Controller besitzen eine einzigartige nicht veränderbare Seriennummer. Diese kann mit diesem Befehl abgefragt werden.

Beispiel:

```
>AT!00  
12345-67890-123
```

>

>AT!01

Ausgabe des Controllertyps und der Bios-Versionsnummer. Da der Identifikations-String veränderbar ist, ist keine eindeutige Identifizierung des Controllertyps und der Bios-Version über den ATI Befehl möglich. Aus diesem Grund wurde dieser Befehl eingefügt, dessen Ausgabe unveränderlich ist und immer den richtigen Controller-Typ anzeigt.

Beispiel:

```
>AT!01  
AGV2055-40
```

>

Obiges Beispiel zeigt die Ausgabe eines AGV2055 Controllers mit Bios-Version 4.0

Weitere Befehle, speziell für die Änderung des Identifikationstextes und zum internen Test des AGV-Interfaces bei der Herstellung werden hier nicht aufgeführt. Interessenten können diese Informationen unter Angabe des Verwendungszwecks beim Entwickler der AGV-Controller per E-Mail anfordern.

Direkteingabe eines OBD2-Befehls

Zur Eingabe eines OBD2-Befehls werden nur die für diesen Befehl notwendigen Daten im Hexcode übergeben. Sollte zuvor noch keine Verbindung mit dem OBD2-Bus im Fahrzeug aufgebaut worden sein, wird dies bei Eingabe des ersten Befehls einmalig vorgenommen. Die verschiedenen Protokolle werden in folgender Reihenfolge durchgetestet:

1. ISO9141/KWP2000 Slow-Init
2. KWP2000 Fast-Init

Sobald der Fahrzeugcontroller eine Antwort bei Anfrage mit einem der Protokolle liefert, wird der weitere Test abgebrochen und das erkannte Protokoll auch für alle weiteren Anfragen benutzt. Die Dauer bis zur Ausführung des ersten Befehls dauert somit maximal 2,5 Sekunden, bedingt durch die lange Antwortzeit beim Slow-Init.

Der K/KL-Transit-Modus

Im K/KL-Transit-Modus wird das Interface in den Transparent-Modus geschaltet und nur zur Pegelwandlung zwischen RS-232-Anschluß und OBD-Stecker benutzt. Das Interface befindet sich in einem passiven Modus, in dem alle Daten vom PC geliefert werden müssen. Der K/KL-Transit-Modus wird aktiviert, indem beim Einstecken des Interface auf den OBD2-Anschluß der Taster an Pin 9 (KEY auf GND) gedrückt und festgehalten wird.

Beispiel:

Ausgabe bei ISO9141 oder KWP2000 Connect.

```
>0100  
BUS INIT: ...OK  
41 00 E8 19 30 12
```

>

```
>0100  
41 00 E8 19 30 12
```

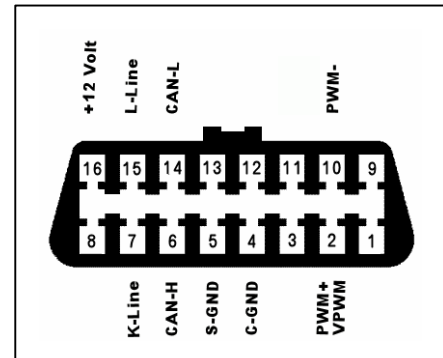
Angezeigt wird der Modus durch ständiges Leuchten der LED1 (Grün).

Der K/KL-Transit-Modus funktioniert nur direkt an der seriellen Schnittstelle oder mit einem USB/Seriell-Wandler mit FTDI-Chipsatz. Andere USB/Seriell-Wandler sind nicht geeignet, da sie die extrem langsamen Baudraten (5 Baud) für die Initialisierung nicht übertragen können.

Die OBD2-Anschlußbuchse

Alle neu hergestellten Fahrzeuge müssen über solch eine 16-polige Buchse nach OBD2-Norm verfügen. Die Buchse muß sich in 1 Meter Umkreis vom Fahrersitz befinden. Obwohl die Bauform genormt ist, benutzen die verschiedenen Fahrzeughersteller verschiedene Übertragungsprotokolle.

Genormt ist jedoch, welches Protokoll an welchem Pin der Buchse anliegt. In den meisten Fahrzeugen sind nur die Kontakte der 16-poligen Buchse vorhanden, die auch Signale führen. Oft sind zusätzliche (nicht genormte) Kontakte an Pins vorhanden und auch mit dem Steuergerät verbunden, die zur internen Diagnose der Fahrzeughersteller dienen. Da es für diese Kontakte keine offiziellen Dokumentationen gibt, sollte man diese möglichst auch immer unbelegt lassen.



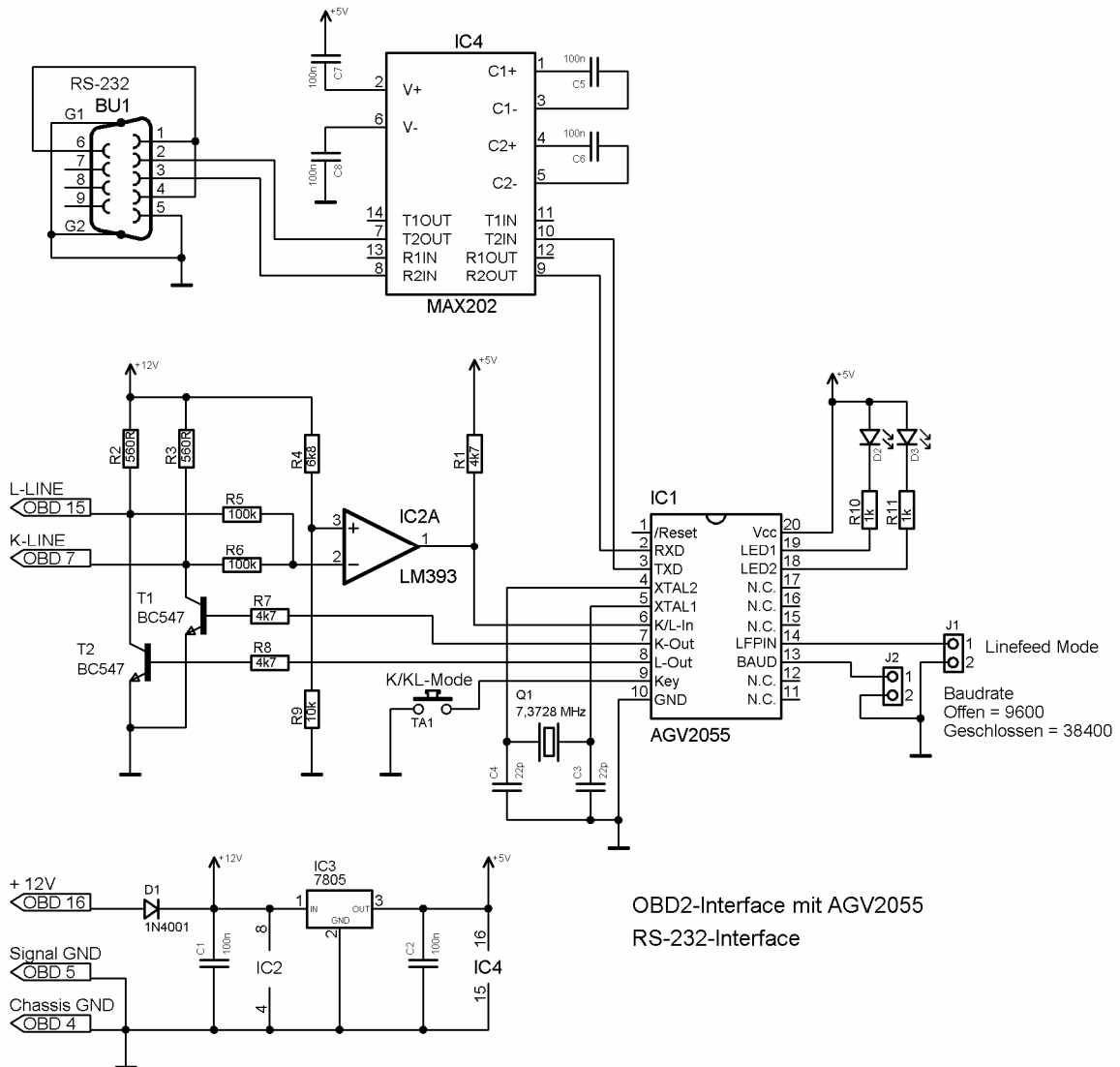
Folgende Kontakte sind immer vorhanden:

PIN 4	Fahrzeug-Chassis-Masse	(Pin 4 + 5 sind meist verbunden)
PIN 5	Signal-Masse	
PIN 16	Stromversorgung + 12 Volt	

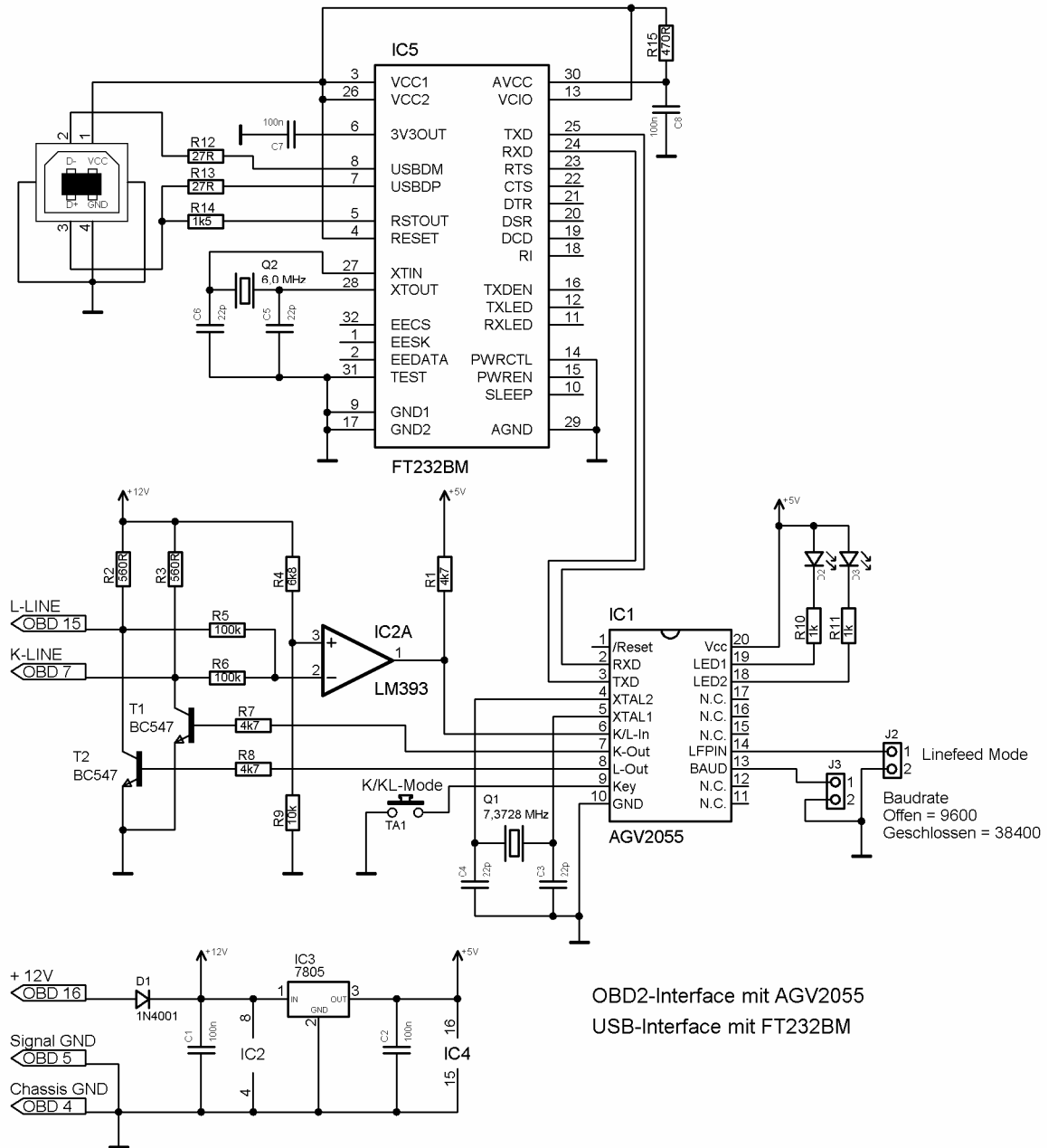
Folgende Kontakte sind je nach verwendetem Protokoll vorhanden und belegt:

PIN 7	ISO9141 / ISO14230 K-Line	(Ein- und Ausgang)
PIN 15	ISO9141 / ISO14230 L-Line	(Eingang, Ein- und Ausgang bei VAG)
PIN 2	J1850, PWM+	(Differentialsignal Bi-Direktional)
PIN 10	J1850, PWM-	(Differentialsignal Bi-Direktional)
PIN 2	J1850, VPWM	(Bi- Direktional)
PIN 6	ISO15765, CAN-H	(Differentialsignal Bi-Direktional)
PIN 14	ISO15765, CAN-L	(Differentialsignal Bi-Direktional)

Für das OBD2-Interface mit AGV2055 werden nur Masse (Pin 4 und 5) die Stromversorgung (Pin 16) sowie K-Line und L-Line (Pins 7 und 15) benötigt.



OBD2-Interface mit AGV2055
RS-232-Interface



OBD2-Interface mit AGV2055
USB-Interface mit FT232BM

Hinweise

© Erwin Reuß; Folker Stange. Nutzung und Weitergabe dieser Informationen auch Auszugsweise nur mit Erlaubnis der Copyright-Inhaber.

Änderungen am Bios und Befehlssatz des Controllers vorbehalten.

OBD-Diag® ist eingetragenes Warenzeichen.

Alle Markennamen, Warenzeichen und eingetragenen Warenzeichen sind Eigentum Ihrer rechtmässigen Eigentümer und dienen hier nur der Beschreibung.

Haftungshinweis

Der Hersteller übernimmt keine Haftung für Schäden die durch Anwendung der OBD-DIAG Interface und der Diagnose-Software entstehen könnten.

Kontakt, Forum

<http://www.obd-diag.de>