» Main Menu

Alliance Membership
Alliance DIY Index

Forums
· Today's Posts
· NAMLive

Marketplace

Reviews
· Products (**351**)

Clubs NEW
· Club Directory (**214**)
· Club & Regional Forums
· Club Services (FREE)
· Add Your Club

NAM Calendar
NAM VIN Decoder
NAM Screensavers

FAQs
Archive

Vendor Directory

Site Guidelines
Contact Us

Getting more out of the R53 MFSW

Post Reply

Page 2 of 3 < 1 **2** 3 >

Like Share 0 +1 0 Tweet 0

Thread Tools | Search this Thread

10-01-2008, 03:05 PM #26

GBMINI
Vendor
ALLIANCE

Join Date: May 2002
Location: Gloucester, MA, USA
Posts: 3,433
Feedback Score: 0 reviews
Gallery

You're exactly correct about OUT10...13 and the V1 signals.

The only issue is if the controlled device has switch signals which are not pulled to ground to trigger - but then only if the device is also connected to the car wiring (eg to power it). In that case, you'd need relays, but otherwise the "open collector" output you have is perfectly fine.

Ian C. Gloucester, MA, USA (MINIless!)
(GBMINI.net, GPMINI.net)

Quote

» Latest Main Topics

Another legitimate Oil Catch Can question...
by bikerbob951
Drivetrain (Cooper S)
115 Replies, 9,509 Views

What did you do to your mini today?
by itsminibtch
General MINI Talk
15,142 Replies, 1,345,618 Views

Pothole!
by sbninja
Tires, Wheels, & Brakes
0 Replies, 1 Views

Your ULTIMATE R53 Newbie, DIY, and Performance Guide
by shorty_du_op
R50/R53 :: Hatch Talk (2002-2006)
110 Replies, 117,089 Views

New Clubman Owner
by JohnDoe11
R55 :: Clubman Talk (2008+)
7 Replies, 419 Views

R56 Battery registry
by rtbship
R56 :: Hatch Talk (2007+)
12 Replies, 627 Views

ENGINE SPUTTER IN 4th,5th,6th gear on hard acceleration
by mydime13
R56 :: Hatch Talk (2007+)
15 Replies, 616 Views

Help , half engine light is on , car now struggles doing...
by MiniTech101
R56 :: Hatch Talk (2007+)
2 Replies, 122 Views

Another SZL failure, cleaning did not help
by erjon
R56 :: Hatch Talk (2007+)
38 Replies, 2,073 Views

rota grid V
by majestic
R50/R53 :: Hatch Talk (2002-2006)
1 Replies, 404 Views

10-01-2008, 05:48 PM #27

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Ian, thanks for taking the time to go over the circuit and the finer points of this stuff.

> Quote:
> Originally Posted by **GBMINI**
> *You're exactly correct about OUT10...13 and the V1 signals.*

Whew, I was worried for a second that I was gonna blow something up...

Great, I'll keep it that way for the V1 signal. Actually now that you've pointed this out, it dawns on me that in this case, since Out is tied to the V1's GND which is the same as the car's GND...I guess I was doing it right, though not due to any actual skill on my part

Trial and error got me to a working solution (it didn't work when hooked up the other way), but it's nice to have an explanation for why things are.

> Quote:
> Originally Posted by **GBMINI**
> *The only issue is if the controlled device has switch signals which are not pulled to ground to trigger - but then only if the device is also connected to the car wiring (eg to power it). In that case, you'd need relays, but otherwise the "open collector" output you have is perfectly fine.*

I'll have to run a meter on the leads from the iPod connector (which does share power/ground with the car) as they seem to work as hooked up no matter which wires I matched to the collector or emitter. Orientation doesn't seem to matter at all. Might just be a fluke. I left some room at the top right of my board for additional stuff like relays so I have room to throw some on if needed.

Yet another reason why I personally wasn't about to go commercial and sell these circuits...I know the design works for me, but haven't the expertise to make sure it is a generalized/safe/foolproof enough solution to actually sell.

BTW, I don't know if you've delved into coding for the Arduino (truthfully, I'm a better coder than a hardware guy) yet but it's pretty darned easy to write code for and quite nice. Certainly easier than writing for the AVR or some other PIC in Assembly. I'll post my code once I've had a chance to proof it/comment it and do some slight tweaks/optimizations on it.

Thanks again,

Dave

_____
"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Quote

10-01-2008, 06:00 PM #28

GBMINI
Vendor
ALLIANCE

Join Date: May 2002
Location: Gloucester, MA, USA
Posts: 3,433
Feedback Score: 0 reviews
Gallery

Hadn't heard of Arduino before - I guess since I tend to "roll my own" as it were! Just took a quick peek, looks pretty neat; the language is interesting, a sort of Basic / C cross. I created a similar language back in 1995(!) to run our company test rigs.

And I agree about PIC assembler, it's a pain - I use C for that processor family (but I can still code direct to HEX for the 6502!)
The HKenabler and iBus remote are in HC908 assembler, just because I did my initial development on such a board (history on GBMINI.net)

I look forward to seeing where you take this project, next

_____
Ian C. Gloucester, MA, USA (MINIless!)
(GBMINI.net, GPMINI.net)

Quote

10-01-2008, 06:31 PM #29

Join Date: Sep 2006
Location: Holly Springs, NC

BlimeyCabrio
6th Gear

Posts: 8,768
Feedback Score: 5 reviews
Gallery

Ian, you're a geek.

Just in case you didn't know.

Takes one to know one.

I'm Paul, The car is Blimey--- BlimeyCabrio's Blog--- 2006 MCSCa w/a few mods and Union Jacks

Thirteen-time Dragon Veteran with Switchback Sixth Sense, hasn't looked like that in a while...

Quote

10-02-2008, 02:04 AM #30

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Quote:

> Originally Posted by **GBMINI**
> *And I agree about PIC assembler, it's a pain - I use C for that processor family (but I can still code direct to HEX for the 6502!)*

Oh man, that brings back memories of my geek days from high school. I remember that I couldn't afford to buy an Assembler Software for the Apple II, but you could just enter "CALL -151", enter your opcodes (I think I had like 50% of the opcodes for the 6502 instruction set committed to memory) in Hex by hand and run your program right then and there, no compiling, no linking, nada...oh those were the days.

That's OK Ian, I'm geeking out big time, right with you.

Dave

"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Quote

10-03-2008, 04:37 PM #31

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Program Code for the IBUS-Arduino circuit

They call programs for the Arduino's "Sketches". Go to www.Arduino.cc to download the free development environment (a C like language, has versions that will run on Windows, OS X and Linux). It's not quite as powerful as programming in AVR assembly, but MUCH easier with lots of external libraries to expand the AVR's capabilities w/o you having to re-invent the wheel and write the code yourself (ie. Math Libraries, I2C serial communications, LCD Displays, Servo Controllers, etc).

Here is a basic sketch that listens for the Tel button and the R/T button to be pushed and will control 2 external devices based on which buttons are pushed.

As it is written, it's not a particularly elegant algorithm and there are some problems with occasionally missed commands and slow response if too many button pushes are stacked up in a row, but it works.

I have a a newer version that is much more elegant, responsive and flexible that I'm still debugging/testing but ultimately I want to write a version that will use interrupts so that no button pushes will ever be missed and multiple complex sequences of commands for the outputs can be executed w/o any dropped incoming IBUS data/commands.

Anyways, this may be a good start for DIY'ers to play with and for experimenting.

Dave

Quote:

```
*/
```

```
IBUS-Arduino Driver v1

Copyright 2008 David Chen (JCW@DaveChen.org) All Rights Reserved
Released under the Creative Commons Attribution Non-Commercial Use License
http://creativecommons.org/licenses/by-nc/3.0/

*/

//main serial Rx pin switched via NPN transistor, this pin is connected to the base of the transistor
//serial Rx port is effectively blocked until activateSerialPin is set to high
//serial data from IBUS interface will keep Arduino from fully starting up otherwise
#define activateSerialPin 2

//this version switches 2 devices, a Radar Detector Mute and a 2 button Garage Door opener
#define garage1Pin 10
#define garage2Pin 11
#define radarmutePin 12


//diagnostic LED light, strobes while getting data and stays lit during matches
#define ledPin 13

//Initialize the Output Pins and the Serial communications with the IBUS Module
void setup() {
pinMode(garage1Pin, OUTPUT);
pinMode(garage2Pin, OUTPUT);
pinMode(activateSerialPin, OUTPUT);

pinMode(ledPin, OUTPUT); //we'll use the debug LED to output a heartbeat

//start recieving Serial Data
Serial.begin(9600);
digitalWrite(activateSerialPin, HIGH);
}

//Main Program Loop
void loop() {
char incomingByte;
char sequenceBytes[4];

// Look to see if there is any IBUS data to process
if (Serial.available() > 0) {
digitalWrite(ledPin, HIGH); //light the debug LED
// read the incoming byte:

incomingByte = Serial.read();

//check if the incoming byte fits our desired patterns
if ((incomingByte == 80) || (incomingByte == -48)) {
lcd.cursorTo(2,0); //Show Match result on right half of the screen
//grab the next 5 bytes
if (Serial.available() > 0) {
sequenceBytes[0] = Serial.read();
}
if (Serial.available() > 0) {
sequenceBytes[1] = Serial.read();
}
if (Serial.available() > 0) {
sequenceBytes[2] = Serial.read();
}
if (Serial.available() > 0) {
sequenceBytes[3] = Serial.read();
}
if (Serial.available() > 0) {
sequenceBytes[4] = Serial.read();
}

if(sequenceBytes[0] == 4) {
switch (sequenceBytes[3]) {
case 16:
// Vol Down
break;
case 17:
// Vol Up
break;
case 1:
// Right
break;
case 8:
// Left
```

```
break;
case 0x80:
//Tel button
// Garage 1 button pushed
digitalWrite(garage1Pin, HIGH);
delay(1000);
digitalWrite(garage1Pin, LOW);
break;
case -112:
// Tel Hold
break;
}
}
else if ((sequenceBytes[0] == 3) && (sequenceBytes[2] ==1)) {
// R/T button
// V1Mute and Garage 2
digitalWrite(radarmutePin, HIGH);
digitalWrite(garage2Pin, HIGH);
delay(500);
digitalWrite(radarmutePin, LOW);
delay(500);
digitalWrite(garage2Pin, LOW);
}
}

}
digitalWrite(ledPin, LOW);
} //end loop
```

"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Last edited by DaveC; 10-03-2008 at 06:39 PM.

Quote

07-09-2009, 01:00 PM #32

Maarten CH
Neutral

Join Date: Jul 2009
Posts: 6
Feedback Score: 0 reviews
Gallery

Thanks for posting this interesting stuff. I am searching for something to use the MFSW buttons to control my 2nd gen Ipod touch.

I found a lot about the Arduino but I still have several questions

Can you explain (in easy words) what the Resler Ibus interface is doing? It is reading the Ibus commands but what type of signal (digital?) is coming out? And what is then read by the Arduino?

It would be great if you can give a link/document with more info about the Resler

Many thanks!
Maarten

Quote

07-10-2009, 01:29 AM #33

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

The Resler I-Bus module is doing the actual interfacing with the actual I-Bus and monitoring/spitting out serial data of what is flowing across the I-Bus.

The Arduino reads the serial data from the I-Bus and looks for strings of data that matches the 5-6 byte pattern that corresponds to a particular event (ie. a MFSW Button push) and then acts on it (ie. closes a switch or relay).

Dave

"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Quote

07-10-2009, 01:39 AM #34

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Info on the Reslers I-Bus interface can be found here: http://reslers.de/IBUS/index.html

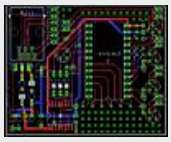
I had gotten around to making a custom circuit board (picture of the board below) that used the Melixis chip directly (thus didn't need the Reslers I-Bus interface) with an Arduino Nano on board, but alas, I'm a bit of an amateur and it isn't working as expected (no I-Bus serial data coming out at all). Not sure what the problem is, but I can post the schematic/design if someone wants to help me trouble shoot it and sort it out.

Basically, I know just a bit about digital electronics and a little more about coding the software. I took bits and pieces of Rolf Reslers and other folk's designs (from the Arduino web site tutorials) to make this board, had to teach myself how to use Eagle to design a board and submitted a design to get made. It was fun, but very time consuming.

Actually, I haven't worked on this stuff in a few months, I was figuring out stuff by trial and error on my own time and have hit a bit of an impasse. If there are others out there who'd be interested in collaborating be more than happy to share what I've figured out already to see if improvements can be made in the design.

Dave

Attached Thumbnails



"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Last edited by DaveC; 07-10-2009 at 02:03 AM.

Quote

07-10-2009, 03:25 AM #35

AKIndiMini
6th Gear

Join Date: Oct 2006
Location: Kodiak, AK
Posts: 7,888
Feedback Score: 0 reviews
Gallery

Does anyone know if the 'extra button' on the 2-spoke MFSW sends an event to the I-Bus?

I kinda like my 2-spoke steering wheel and it would be great if I could make that extra button 'do' something, like open my garage door...

Darren Asuncion
Kodiak, Alaska

04 MCS IB/W - Indi Blue Owner's Club #76, MMC#515
Flickr Facebook YouTube Myspace

Quote

07-10-2009, 06:59 AM #36

Maarten CH
Neutral

Join Date: Jul 2009
Posts: 6
Feedback Score: 0 reviews
Gallery

So the Arduino can't read the Ibus itself and needs the Resler to do that.
Sounds ok but how did you figure out which pins from the serial output of the Resler to take?

I was thinking of taking the serial Resler (I think is easier to connect than the USB version) and connect it to the Arduino Duemilanove (http://arduino.cc/en/Main/ArduinoBoardDuemilanove).
Other/better ideas are

Quote

07-10-2009, 07:00 AM #37

GBMINI
Vendor
ALLIANCE

Join Date: May 2002
Location: Gloucester, MA, USA
Posts: 3,433
Feedback Score: 0 reviews
Gallery

Yes, it does. This, for example, was triggered by the unused middle button:
http://www.gbmini.net/wp/2004/09/ano...dd-on_circuit/

And this earlier button brought out all the controls, for a friend to wire in to his iPod remote (in the days before easy iPod control was available in cars!)
http://www.gbmini.net/wp/2004/06/mor...ol_using_mfsw/

Ian C. Gloucester, MA, USA (MINIless!)
(GBMINI.net, GPMINI.net)

Quote

07-10-2009, 07:06 AM #38

GBMINI
Vendor
ALLIANCE

Join Date: May 2002
Location: Gloucester, MA, USA
Posts: 3,433
Feedback Score: 0 reviews
Gallery

One other note ...
If you're doing READ ONLY iBus monitoring (ie: don't need to send messages out), you really need nothing more than a resistor (and maybe a transistor) to interface between the iBus and a processor - the circuit pictured top of this page has just a resistor (and smart programming); the transistor is there to drive the relay output, not to interface to iBus.

Ian C. Gloucester, MA, USA (MINIless!)
(GBMINI.net, GPMINI.net)

Quote

07-10-2009, 01:33 PM #39

AKIndiMini
6th Gear

Join Date: Oct 2006
Location: Kodiak, AK
Posts: 7,888
Feedback Score: 0 reviews
Gallery

> Quote:
> Originally Posted by **GBMINI**
> *Finally, if there was enough interest in this, I might be able to produce a circuit that connected to iBus and brought transistor outputs to mimic the switches, for certainly less than the $90 price you needed to pay for your parts - but there would need to be more than a couple of people needing it, to bring the price down. Not that I'm wanting to steal anything, it's just that I've done it before ...*

Ian - I'm most definitely interested in this. Hopefully others will be too. Thanks!

Darren Asuncion
Kodiak, Alaska

04 MCS IB/W - Indi Blue Owner's Club #76, MMC#515
Flickr Facebook YouTube Myspace

Quote

08-02-2009, 12:45 PM #40

Maarten CH
Neutral

Join Date: Jul 2009
Posts: 6
Feedback Score: 0 reviews
Gallery

Inspired by the nice example from Dave I started a project as well. I have an Arduino and the Resler interface should come in the next few days

In post #9 you wrote that you used Rx (pin 9), Tx (pin 10) and CTS (pin 11) from the Resler interface. Is it correct that these are pin 2, 3 & 8 at the serial connector? Some time ago you posted the code you wrote for the Arduino.

Do you have a more resent version where you also send data?

I will keep you updated about my progress and maybe there are still some questions to come.

Cheers,
Maarten

Quote

08-07-2009, 04:31 AM #41

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

> In post #9 you wrote that you used Rx (pin 9), Tx (pin 10) and CTS (pin 11) from the Resler interface. Is it correct that these are pin 2, 3 & 8 at the serial connector?

Yes, the pins 9, 10, 11 are TTL signals that I intercept to send to the Arduino BEFORE they pass through the serial comm chip (MAX232) which puts out a Serial signal at pins 2, 3 and 6/8. The Serial signals are different enough (different Line Levels and voltages) that I didn't try to interface them to the Arduino who's serial pins read TTL signals.

> Some time ago you posted the code you wrote for the Arduino. Do you have a more resent version where you also send data?

I haven't gotten around to sending data across the I-BUS. I think what you need to do that is listen for pin 11 to go low to indicate that the I-BUS is clear for incoming signals before sending data/message over pin 10. Haven't messed with this but I was intending to do it at some point. If you can get it to work, I'd be interested to see how you did it and to see if the theory on how it is to be done (via information kindly provided by Rolf Resler) works.

> I will keep you updated about my progress and maybe there are still some questions to come.

That would be great, thanks for sharing your experiences. As I've said in the past, the project has been at a stand still for me as I've not had much time recently to mess with it.

I'm still trying to get an Arduino to work without the Resler interface (fewer components the better). Ian, I think you mentioned that you can interface with the I-Bus for data reading purposes w/o the Melixis/I-Bus interface circuitry. Would you mind sharing a schematic of how/what kind of resistor/transistor would be needed to do this?

Thanks all for taking an interest and sharing information,

Dave

"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Quote

08-07-2009, 04:47 AM #42

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Improved Arduino Code for reading MFSW codes

This is the improved code I mentioned before for grabbing MFSW codes.

Rather than repetitive pattern matching for particular MFSW button push byte sequences, this newer version looks for the byte pattern "80" that is the start of the data stream that indicates a MFSW button push event.

The program then reads the next 3 bytes and calculates a checksum (a rather arbitrary calculation that creates an unique checksum for each button event, but since it is just addition/subtraction it is quite a fast calculation. I suppose I could have used XOR's and bit shifting, but this was quick and dirty).

A "case" command is used to implement the action required for each particular button push.

Code:

```
/* Match Patterns from the I-Bus/K-Bus for the MFSW Button Pushes Version 2
*/

/* Licensed for FREE use under Creative Commons Attribution-Noncommercial 3.0 United State

/*
 main serial Rx pin switched via NPN transistor, this pin is connected to the base of the
 serial Rx port is effectively blocked until activateSerialPin is set to high
 serial data from IBUS interface will keep Arduino from fully starting up otherwise
*/
#define activateSerialPin 2
#define CTSSerialPin 3

#define ipodKeyPress 50    // hold the ipod control button for this many msec
#define fwdipodPin 10
#define playipodPin 11
#define revipodPin 12

#define radarKeyPress 100
#define radarmutePin 13

// Extra Pin Definitions - not hooked up to hardware but can be assigned as needed
#define freeKeyPress 0
#define freePin4  4
#define freePin5  5
#define freePin6  6
#define freePin7  7
#define freePin8  8
#define freePin9  9

void setup() {
  pinMode(activateSerialPin, OUTPUT);
```

There is an additional optimization that I want to implement at some point. Currently, the code uses a pre-defined delay between toggling the respective control pins high and low. A more elegant solution is to set a array element (corresponding to a particular output pin) to the Arduino's current time tick/counter value. A second check would run through the array and see when the current time tick/counter is greater than a defined value more than the array element's value and flip the pin's state. This way, the main loop isn't stuck in a delay command while waiting to toggle a pin state and the whole program will be (potentially) more responsive.

I'm sure my description is less than clear, this code example may be a better description of what I mean to do:
http://www.arduino.cc/en/Tutorial/BlinkWithoutDelay

Feel free to use and or improve the code and share at will. I'm not a trained programmer, just a hobbyist and I'm sure that someone out there can come up with much more elegant code.

Regards,

Dave

_____
"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Quote

08-07-2009, 04:52 AM #43

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Quote:

> Originally Posted by **GBMINI**
> *One other note ...*
> *If you're doing READ ONLY iBus monitoring (ie: don't need to send messages out), you really need nothing more than a resistor (and maybe a transistor) to interface between the iBus and a processor - the circuit pictured top of this page has just a resistor (and smart programming); the transistor is there to drive the relay output, not to interface to iBus.*

Ian, would you be willing to share a Schematic for how you'd interface w/o the I-Bus interface? What resistor/transistor values would you suggest.

It would be a great help, especially since you actually are an electronics expert and know what you're doing. I'm a bit of a hack/amateur at this and have been going at it via trial and error hoping not to fry anything important.

Thanks,

Dave

"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Quote

08-07-2009, 05:38 AM #44

BlimeyCabrio
6th Gear

Join Date: Sep 2006
Location: Holly Springs, NC
Posts: 8,768
Feedback Score: 5 reviews
Gallery

I'd buy a preconfigured circuit that does this stuff in a heartbeat, if such a thing existed (anymore). I'm loving what y'all are doing - I just don't have the time or energy to DIY this anytime soon. So, Ian, put me on the list...

I'm Paul, The car is Blimey--- BlimeyCabrio's Blog--- 2006 MCSCa w/a few mods and Union Jacks

Thirteen-time Dragon Veteran with Switchback Sixth Sense, hasn't looked like that in a while...

Quote

08-07-2009, 07:01 AM #45

GBMINI
Vendor
ALLIANCE

Join Date: May 2002
Location: Gloucester, MA, USA
Posts: 3,433
Feedback Score: 0 reviews
Gallery

Dave,
The key is to be able to get the software working OK.

The "interface" to read iBus data is no more than a single resistor! Something like a 22K or 33K between the iBus and an input of the "PIC"
Although the iBus goes to 12V, the processor has built-in voltage limit diodes, so the "spare" voltage just drops across the resistor - and choosing a high value resistor limits the current so you don't hurt the input pin or the iBus itself.
I used 33K on my HKenabler circuits.

The key however, is that without any transistor interface, the iBus data is "upside down" at the processor. From my s/w comments:

> Quote:
> ... communicates with a Harmon Kardon amplifier across the MINI iBus, which is a serial bus running at 9600/E/8/1, using open-collector output to transmit.
> The bus is non-inverted, high when idle with data beginning with a low start bit ...

If you use a built in "UART" (serial interface) electronics in the "PIC", it won't be able to decode the data.
So instead you need a software serial receive routine, which handles the "upside down" pin. That was easy on my project since I just wrote the code. I don't know if the Arduino offers that facility - though I bet it can.

Ian C.

> Quote:
> Originally Posted by **DaveC**
> *Ian, would you be willing to share a Schematic for how you'd interface w/o the I-Bus interface? What resistor/transistor values would you suggest*

Ian C. Gloucester, MA, USA (MINIless!)
(GBMINI.net, GPMINI.net)

Quote

08-07-2009, 09:29 AM #46

DaveC
2nd Gear

Join Date: Oct 2004
Posts: 107
Feedback Score: 2 reviews
Gallery

Ian, thanks for the info. Sounds like it's doable but for the Arduino as there is a "software serial" library/routine you could probably adapt to account for the signal inversion, but there is no buffering available via this method. Though ideally, it'd be nice to use one of the UART driven serial inputs as that way, there is some buffering of the serial data so no bytes/data are lost.

If I'm reading this right, it sounds like a 33K resistor to a Transistor or some other way to change the polarity of the signal (from 0<->Negative voltage to 0<->Positive voltage) is in order. Not sure if the term "Pull Up" circuit applies in this case. Any suggestions on hooking up a Transistor to accomplish this? Maybe something like this attached image? Not sure what appropriate Resistor values should be used to convert an inverted 12V serial signal to a 5V TTL signal (needed by the Arduino).

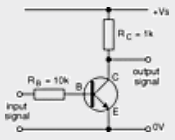
I think I'm in way over my head on this one.

Dave

Quote:

> Originally Posted by **GBMINI**
> *Dave,*
> *The key is to be able to get the software working OK.*
>
> *The "interface" to read iBus data is no more than a single resistor! Something like a 22K or 33K between the iBus and an input of the "PIC"*
> *Although the iBus goes to 12V, the processor has built-in voltage limit diodes, so the "spare" voltage just drops across the resistor - and choosing a high value resistor limits the current so you don't hurt the input pin or the iBus itself.*
> *I used 33K on my HKenabler circuits.*
>
> *The key however, is that without any transistor interface, the iBus data is "upside down" at the processor. From my s/w comments:*
>
> *If you use a built in "UART" (serial interface) electronics in the "PIC", it won't be able to decode the data. So instead you need a software serial receive routine, which handles the "upside down" pin. That was easy on my project since I just wrote the code. I don't know if the Arduino offers that facility - though I bet it can.*
>
> *Ian C.*

Attached Thumbnails



"OX" 2006 MCS SB/S Factory JCW, LSD, JCW Brakes w/ SS Brake Lines, JCW Wing, Sport Suspension w/ H-Sport Rear Sway Bar, Alta Rear Lower Control Arms, Powerflex suspension Bushings, RMW Tune, RMW Cam, RMW Shorty Exhaust Header, Clutchmaster Stage 4 Clutch and Lightened Flywheel, Brisk Spark Plugs => 215 WHP/186 TQ

Last edited by DaveC; 08-07-2009 at 09:38 AM.

Quote

08-07-2009, 10:53 AM #47

GBMINI
Vendor
ALLIANCE

Join Date: May 2002
Location: Gloucester, MA, USA
Posts: 3,433
Feedback Score: 0 reviews
Gallery

That attached schematic is a perfectly usable transistor inverter.
The input resistor could be 10K, the resistor above is fine at 1K, but going in to the "PIC" input another 10K would also be perfect.

The transistor wants to be any inexpensive NPN signal transistor; we use 2N4401 or similar, but basically if it's good gain and not slow, it'll work.

Ian C. Gloucester, MA, USA (MINIless!)
(GBMINI.net, GPMINI.net)

Quote

08-07-2009, 11:19 AM #48

jbkone

Join Date: Dec 2008
Location: Pensacola, FL
Posts: 348
Feedback Score: 0 reviews

4th Gear

Gallery

I'm barely following... and only 50% of that! But fascinating! Keep it up.

Quote

08-08-2009, 08:15 PM #49

blalor
6th Gear

Join Date: Jan 2005
Location: Medford, MA
Posts: 1,224
Feedback Score: 0 reviews
Gallery

Had a few minutes to kill tonight and Googling "arduino ibus" landed me here, via the arduino.cc forum. Arduino/IBus integration has been on my list for a while, now. I originally gave the MCAW guys a $100 deposit for that IBus interface circuit they were developing, and that didn't end well. I pretty much just want to do the same thing that Dave is doing: mute my V1, and also turn it off for a period of time (like when I'm in a shopping center and there's a new bogey popping up every 3 seconds). I've been into the Arduino for a while and have been working on some "smart home" stuff, but haven't gotten around to building the interface circuit for my MINI, yet. I also want to do some OBD2 datalogging with an ELM323...

Ian, if you're looking for another potential customer for a general-purpose IBus circuit, sign me up!

Quote

08-26-2009, 03:08 PM #50

Maarten CH
Neutral

Join Date: Jul 2009
Posts: 6
Feedback Score: 0 reviews
Gallery

> Quote:
>
> Yes, the pins 9, 10, 11 are TTL signals that I intercept to send to the Arduino BEFORE they pass through the serial comm chip (MAX232) which puts out a Serial signal at pins 2, 3 and 6/8. The Serial signals are different enough (different Line Levels and voltages) that I didn't try to interface them to the Arduino who's serial pins read TTL signals.

Thanks for the info! I will go for that as well.

Here is a small update:

Last weekend I soldered the cables to the connector in the back of the radio (no CD changer connector to use) and the Reslers IBus interface is working with my laptop. It can read and send data to the Ibus. It didn't work with the Arduino so far but I will give it another try soon.

However, my goal is to control my Ipod touch in it's IR controlled docking station (that needs to be build in the car) and I made some progress on this side . I decoded the Apple IR remote and can send all the commands with the Arduino using just a single IR LED. With this solution the docking station can be used in the trunk or glove box and I don't have to open/kill the original remote. I can post the codes if someone is interested.

I hope to solve the last issues this weekend

Quote

08-26-2009, 03:08 PM

Post Reply

« Previous Thread | Next Thread »

Posting Rules

You **may not** post new threads
You **may not** post replies
You **may not** post attachments
You **may not** edit your posts

BB code is **On**
Smilies are **On**
[IMG] code is **On**
HTML code is **Off**
Trackbacks are **Off**
Pingbacks are **Off**
Refbacks are **Off**

Forum Rules

Forum Jump
Electrical Go

MINI Cooper news, forums, FAQs, and reviews for enthusiasts and owners of the North American MINI Cooper All times are GMT -7. The time now is 02:04 PM.

All times are GMT -7. The time now is 02:04 PM.