let's make _____ Explore (/tag/type-id/account/Publish (about/create.jsp?sourcea=header) login/account/Publish(about/create/jsp)pro?source=header)

Featured: share what you make(intel/) | Arduino(/tag/type-id/category-technology/channel-arduino/?sort=FEATURED)

(/) Parties(/tag/type-id/category-craft/channel-parties-and-weddings/?sort=FEATURED) | (/halloween/)

Arduino is... SLOW?

(/file/F97ZQOFG145BVESA

Arduino is slow? What? This instructable will show just how slow a part of Arduino is, and how to fix it.

It's true – more specifically, Arduino's digitalWrite command takes a considerable amount of time. If you are just switching on a LED once or something, you won't be able to notice it. However, I realized how slow it was while I was trying to use a TLC5947 PWM driver. That requires the microcontroller to shift in 288 bytes each time! Each byte required about 12 digitalWrites, for a total of 3456 digitalWrites each time I wanted to shift in new data to the TLC5947.

How long did that take? 30 seconds of just digitalWrite!

But there is a solution – using "true c" style commands, or what the AVR GCC (GNU GCC Compiler) uses. The brains behind Arduino are ATMega168s or ATMega328s. The AVR community typically uses "true c" commands to program these chips, using AVR Studio 4. The advantage of using these "true c" commands is that it does exactly what you tell it to do.

But before we get in to these commands, we must get familiar with port and pin definitions in the next step!

(If you predict you will like this instructable, feel free to vote for the Arduino contest!)

Step 1: The Truth about Pins
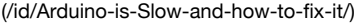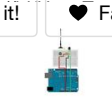
Related

How To Enter the Arduino Contest (/id/How-To-Enter-the-Arduino-Contest/) by Contest Robot

Arduino Electrometer/Capacitive Object Sensor (/id/Arduino-ElectrometerCapacitive-

Arduino EEG (/id/Arduino-EEG/) by Treker2 (/member/Treker2/)

Arduino Robotic Hand with Haptic Feedback

Atmega168 Pin Mapping

Digital Pins 11,12 & 13 are used by the ICSP header for MISO, MOSI, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Arduino users know that the pins are labeled as digital 0-13 and analog 0-5. The makers behind Arduino used this for simplicity. The actual ATMega chip's pins are labeled differently, however. Each pin has an assigned letter and number. The numbers range from 0-7. For example, pins can be A0-A8, B0-B8, and so on. All of the AVR 8-bit pins are labeled this way.

To help you clarify which digital/analog pin corresponds to which AVR pin, see the chart below.

My Seeeduino has a LED built in on digial pin 13. So, looking at the chart, its real pin would be B5.

Next I will show you what the actual C command is.

We have a be nice comment policy. Please be positive and constructive.

🖐 I Made it!   📷 Add Images   Post Comment

jackocurly0074 (/member/jackocurly0074)   3 years ago   Reply

If you count down rather than up in the loop it will be even faster! On my Arduino Mega changing the loop to count down for the True C commands reduced the time taken from 288microseconds to 192 microseconds, big difference if you need it to be as fast as possible!

★ **Kurt E. Clothier (/member/Kurt E. Clothier)** ▸ jackocurly0074

2 months ago   Reply

Counting down is more efficient, but only if you are comparing to zero. It is always more efficient to compare a value to zero than to another value! Similarly, do{}while(); loops are more efficient than for() loops. There are lots of pre-optimization tricks that you can do to squeeze your code into the smallest of chip spaces.

Kurt E. Clothier (/member/Kurt E. Clothier)   4 months ago   Reply

I am glad someone else finally realized the dirty secret behind Arduino... all of those libraries you love you use are horribly inefficient because they have to remap function calls to different pins on different devices.

You also have very little control over how the internal hardware is used (which timer, at what speed, etc). I regularly squeeze my optimized code into 99% of the space available on AVR chips. I want to get everything out of it that I can - no room for fluff.

agr00m (/member/agr00m) ▸ Kurt E. Clothier (/member/Kurt E. Clothier)   Reply

2 months ago

If you know how to program C on an AVR, then it's likely Arduino's are not for you to begin with. However, I regularly program AVRs, but still find Arduino's useful, inefficient or not. The sheer number of libraries for almost every peripheral device imaginable makes it appealing for getting projects quickly off the ground and/or prototyping. If the project provides fruitful, I'll then port it to an AVR or write my own C library for the Arduino.

Kurt E. Clothier (/member/Kurt E. Clothier) ▸ agr00m (/member/agr00m)   Reply

2 months ago

There is nothing wrong with that approach... in fact, I'd say that is probably the most intelligent use of Arudino possible - rapid proof of concept with the intent to optimize the code later.

However, that still doesn't solve the issues many people come across with multiple libraries try take control of the same hardware peripherals or IO Pins, and the user has no idea how any of it works, so they just give up.

zootalaws (/member/zootalaws)   3 months ago   Reply

"Note that | can be found to the left of the backspace key, on the same key of the backslash."

On a Windows PC, with a U.S.-style keyboard.

Different machines will vary.

niels.vandepas.5 (/member/niels.vandepas.5)   8 months ago   Reply

"0b10000000 is an 8-bit binary number, you can convert it to hex for a cleaner look. Doing it manually is a pain"

That's not true, every four bits can be directly translated to hex: 1000 is binary for 8 and 0000 is a 0, so the output will be 0x80. 0b11110101: 1111 is decimal 15 and hexadecimal F, 0101 is binary for 5 (dec and hex). Putting these numbers together gives 0xF5. If you get the hang of it, it is ways faster than looking up every single number.

matstructable (/member/matstructable)   2 years ago   Reply

Thanks for sharing that !

---

rcarvalho4 (/member/rcarvalho4)   4 years ago   Reply

It's better change just the part after the logic operator:

PORTB |= 0b00100000;
PORTB &= ~0b00000000;

If you don't use the logic operators, but just the equal "="
sign, all the pins are going to be set like the byte you sent,
not just pin B5. In the case where you set B5 high, you
would set all ohter B pins to low.

---

n8hfi (/member/n8hfi)   6 years ago   Reply

Part of the reason this works is that the DigitalWrite() and
DigitalRead() do some error checking before it sets the
register bit--in particular, it checks and turns off PWM if it's
enabled for that pin. Directly accessing the bit yourself is
fine, but you're also assuming responsibility for skipping the
check.

This is pretty safe if your sketch is simple and doesn't do
much multitasking or multiplexing of pins, but can get
perilous if you've got a complex one.

---

vilts (/member/vilts)   6 years ago   Reply

 I wonder why I get this error:

In function void loop():
Error: 'PB5' was not declared in this scope.

I use 0018 software.

I used form PORTB = B.....;

and it took 3808 vs 284 microseconds.

I'm trying to get TLC5945 communicating with arduino, no
luck with digitalWrites, hopefully this version works.

---

eewithmac (/member/eewithmac) ▶ vilts (/member/vilts)   6 years ago   Reply

 I have the same problem. The example code is obviously missing
something. Where and how should PB5 be declared?

Me, I'm trying to PWM on all the pins of an Arduino Duemilanove to dim
some LEDs.

RazorConcepts (/member/RazorConcepts) (author) ▶ eewithmac
(/member/eewithmac)   6 years ago   Reply

 Hmm, compiles fine for me on 0015, it may be
something with the new software.

If anyone has problems, replace
" PORTB |= _BV(PB5);
 PORTB &= ~_BV(PB5);"

with:
"PORTB = 0b00100000;
PORTB = 0b00000000;"

Which should toggle PB5 like the original code. PB5
shouldnt have to be declared, as winAVR should

recognize it as a pin, but arduino 0018 may be doing something funky with that.

just_watching (/member/just_watching)   6 years ago   Reply

Thanks this helped a lot becaus its the only way to programm the 16 extra pins on the Seeduino Mega

zuixro (/member/zuixro)   6 years ago   Reply

Thanks for this instructable. Not only does this save a lot of time, it also knocks the program size down quite a bit. You can also you the registers DDRx an PINx (where x is the letter of the port) to change a port's input/output status, and to check what the input value is.

ghaon (/member/ghaon)   6 years ago   Reply

 So do you then use the AVR Studio 4?
Or how do you program an Arduino in this way?
Can you use the Arudino's usb ftdi connection?

Please elaborate. Thanks!

RazorConcepts (/member/RazorConcepts) (author) ▸ ghaon (/member/ghaon)

I use Arduino with the "true c" style commands.   6 years ago   Reply

You can use AVR Studio 4, however it is quite complicated and you cannot use the usb connection, you will need a ISP programmer to burn the programs.

funlw65 (/member/funlw65)   6 years ago   Reply

Maybe using this (http://sites.google.com/site/funlw65/electronics/jaluino-pinguino-28-pins-starting-bo/freejalduino) "Arduino" ?

wholder (/member/wholder)   6 years ago   Reply

Actually, there is a very nice writeup on "Direct Port Manipulation" (which is the Arduino folk's name for what you are doing) on the Arduino site. Here's the link:

  http://www.arduino.cc/en/Reference/PortManipulation (http://www.arduino.cc/en/Reference/PortManipulation)

The description at the top of the page says "Port registers allow for lower-level and faster manipulation of the i/o pins of the microcontroller on an Arduino board."

Wayne

WyoJustin (/member/WyoJustin)   6 years ago   Reply

Nice tutorial. You should enter it into the Arduino contest.

gmoon (/member/gmoon)   6 years ago   Reply

Some explanation of C "bitwise" operators might be helpful, too... And maybe break it down without using compound assignment ops first.

Here's my fav bitwise link: AVR bit tutorial
(http://www.avrfreaks.net/index.php?

Other than that, it's a good 'ible...

RazorConcepts (/member/RazorConcepts) (author) ▸ gmoon (/member/gmoon)

Thanks! That may be a bit complicated, but I added a final step with that link.   6 years ago   Reply

RazorConcepts (/member/RazorConcepts) (author) ▸ RazorConcepts
(/member/RazorConcepts)   6 years ago   Reply

facepalm
horrible pun was unintended

Nerdz (/member/Nerdz)   6 years ago   Reply

How come you didnt you Hardware SPI instead of Bit Banging the commands to the PWM chip? I checked briefly and it seems it use SPI commands. Looking at the Atmega168 Data sheet, It Does have support for hardware SPI, and if you used hardware, it would much Much faster than bit Banging SPI Commands since you can usually get up to 20Mhz Data Transfer depending on your internal clock (or external, but usually it requires you to go with the max clock rate for the chip) since it would Free up the CPU to do other things.

richms (/member/richms) ▸ Nerdz (/member/Nerdz)   6 years ago   Reply

SPI uses the USART doesnt it? And thats pre-wired on the arduino to support the bootloader

RazorConcepts (/member/RazorConcepts) (author) ▸ richms (/member/richms)

True, actually there is a TLC5940 library for Arduino   6 years ago   Reply
that uses SPI, however I wanted to see what was actually going on so I just did it the "manual" way by bit banging.

SPI doesn't use the USART, it uses SS/MOSI/MISO/SCK, the latter 3 are used for ISP programming also.

Dantex (/member/Dantex)   6 years ago   Reply

does this work for arduino mega? (chip is atmega1280

RazorConcepts (/member/RazorConcepts) (author) ▸ Dantex (/member/Dantex)

 Yes it does, however I cannot find a chart that shows   6 years ago   Reply
which arduino pin goes to which pin... you may have to take a peek in to the schematic of the Arduino Mega.

mtbf0 (/member/mtbf0)   6 years ago   Reply

reading and writing pwm pins will also be somewhat slower than writing non-pwm pins.

hmmm, i notice that there's no longer any range checking on pin numbers. although this speeds things up i wonder what havoc a pinMode or digitalWrite to some out of range value might wreak.

flof (/member/flof)

Good to know that there is such an overhead!
How does the ShiftOut() command compare to this?
Looking at the arduino reference it seems to be dedicated
for this task (if the SPI pins are available..)
Would be nice if someone posts some numbers, since i do
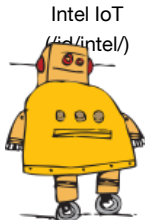not own an arduino to test with..

6 years ago   Reply

westfw (/member/westfw)

There's a pretty extensive discussion on pin-toggling on the
arduino forums here: http://www.arduino.cc/cgi-
bin/yabb2/YaBB.pl?num=1230286016

6 years ago   Reply

FEATURED CHANNELS

Intel IoT (/id/intel/)

Life Hacks (/tag/type-id/category-home/channel-life-hacks/)

Beauty (/tag/type-id/category-home/channel-beauty/)

Woodworking (/tag/type-id/category-workshop/channel-woodworking/)

Minecraft (/tag/type-id/category-play/channel-minecraft/)

Breakfast (/tag/type-id/category-food/channel-breakfast/)

Laser Cut (/tag/type-id/category-workshop/channel-laser-cutting/)

Organizing (/tag/type-id/category-home/channel-organizing/)

Arduino (/tag/type-id/category-technology/channel-arduino/)

Newsletter

Join 2 million + to receive instant inspiration in your inbox.

enter email   I'm in!

Mobile

Download our apps!

Android » (https://play.google.com/store/apps/details?id=com.adsk.instructables)

iOS » (https://itunes.apple.com/app/instructables/id586765571)

Windows » (http://apps.microsoft.com/windows/en-us/app/7afc8194-c771-441a-9590-54250d6a8300)

About Us

Who We Are (/about/)

Advertise (/advertise/)

Contact (/about/contact.jsp)

Jobs (/community/Positions-available-at-Instructables/)

Help (/id/how-to-write-a-great-instructable/)

Find Us

Facebook (http://www.facebook.com/instructables)

Youtube (http://www.youtube.com/user/instructablestv)

Twitter (http://www.twitter.com/instructables)

Pinterest (http://www.pinterest.com/instructables)

Google+ (https://plus.google.com/+instructables)

Tumblr (http://instructables.tumblr.com)

Resources

For Teachers (/teachers/)

Artists in Residence (/air)

Gift Pro Account (/account/give?sourcea=footer)

Forums (/community/)

Answers (/tag/type-question/?sort=RECENT)

Sitemap (/sitemap/)