Literary Clock Made From E-reader

by tjaap

My girlfriend is a *very* avid reader. As a teacher and scholar of English literature, she reads eighty books per year on average.

On her wishlist was a clock for our living room. I could have bought a wall clock from the store, but where is the fun in that? Instead, I made her a clock that tells the time by quoting time indications from literary works, using an e-reader as display, because it's so incredibly appropriate :-)

It updates every minute, so for instance at 9.23 in the evening, the Kindle will read

> My father met me at the station, the dog jumped up to meet me, missed, and nearly fell in front of the 9.23pm Birmingham express.

The way I made this, the Kindle can still be used as a normal e-reader. If the clock is turned on though, as an added bonus, it doubles as a literary quiz. The clock shows the quotation without the title and author of the book, so you can guess. If you want to know the answers, pressing the buttons on the side (normally used to advance pages of e-books) will reveal them.

Update August 5:

Thank you all very much for all the nice compliments! Also, the feedback has been very useful. If you have any trouble making your own Kindle clock, please see the comments.

This Instructable has been featured on Hackaday, Gizmodo, The Verge and Hacker News. I am one very proud and happy maker :-)

Meanwhile, Johannes Enevoldsen made a web version of my clock, as did Davide. I am excited that my project inspired theirs.

Step 1: Tools and Materials

Really the only thing needed is an e-reader (and a USB cable to connect to it). For this project, a Kindle was donated to me by a friend. It is a Kindle 3 WiFi (nicknamed K3, or K3W). You will find many second-hand earlier models like that on eBay for instance.

You'll need a computer (any operating system), with an SSH client like vSSH and an sFTP client like Filezilla installed (both are free). It helps to have a bit of experience with Linux, because that is what the Kindle runs on.

To have the Kindle stand upright in our cupboard, I made a stand from concrete. If you want to do the same, you'll need a food container in a shape you like, cling film, styrofoam, cement, hot glue or double sided tape, and a bucket (to mix the cement).

Step 2: Jailbreaking the Kindle

In order to change the Kindle into a clock, we need to get into the system files. In order to do that, we need to open it up through a process called 'jailbreaking' (don't worry, it's not illegal if it's your property). An explanation to jailbreak the Kindle and a zip file with the necessary files can be found here. Also see this overview of all available software custom software. Find out which Kindle model you have on this page.

For this project, you only need to install the jailbreak hack and the usbnet hack, not the screen saver hack. USBNetwork will grant you remote shell access to your Kindle, either over USB or WiFi. What you will need, if you want to use the keyboard's keys, is the Launchpad hack.

Warning: I read this can potentially ruin your Kindle. Follow the instructions. Jailbreak at your own risk.

your WiFi router and looking it up there. Username is 'root', and the default root password for your model can be calculated.

Then install Python on the Kindle, again using the

If you connect the Kindle to your computer, it will show up as an USB drive.

Basically, all you need to do is put Update_jailbreak_0.13.N_***_install.bin (where *** is your Kindle version, in my case 'k3w') in the root folder of the Kindle when it is connected to your computer.

From the README file in the zip file: "Now, eject & unplug your Kindle, and go to *[HOME] -> [MENU] > Settings -> [MENU] > Update Your Kindle*. It should be quick." (note: that's two times clicking the menu button).

Then do the same for the USBNet and Launchpad files. You should now be able to log in to the device using SSH. On the Kindle, connect to the WiFi network. One way to find out its IP address is by logging into

rel="nofollow">files on the excellent Mobileread.com forum (thanks VoltaX2 in the comments below).

Step 3: Making an Image for Every Single Minute of the Day

There are 1,440 minutes in a day. Compiling a list with quotes for each and every one of them from different literary works is a massive undertaking. Big relief: others already did that for us.

In 2011, newspaper The Guardian asked its readers to submit quotes from books which mention times. They wanted to build an installation for a literary festival. So they have two versions of a list on their website (1, 2).

I combined the two lists, cleaned them up, added a few times I found myself, and turned them into one CSV file.

Unfortunately the list does not cover all minutes of the day. I worked around this by using some quotes more than once, for instance if it can be used both in the AM and PM. More vague time indications can be used around a certain time, so this quote from Catcher in the Rye is used at 9.58AM: "I didn't sleep too long, because I think it was only around ten o'clock when I woke up ... "

Even with this pleasant list, two things took me an unreasonable amount of time. I needed to turn every single quotation from the list into an image. I wanted to make them fit nicely to the screen, so the font would be as large as possible for each quotation.

While scaling a text box to a certain height and width is easy to do manually in most photo editing software, it would have been an immense amount of work to create them one by one. Creating a script to do it for me however proved to be quite the task as well. In PHP (I used that programming language because it has nice functions to deal with text) I wrote a recursive function to find the best fit for each quotation, long or short. For each line, the script creates two PNG images, one with and one without metadata.

It uses the Libertine font, which I like because of its stylish look, because it is very complete (numbers, punctuation, diacritics) and because it's open source.

The other thing that took me a long time is identifying all time mentions in the quotations, because I wanted to write them in bold text. That makes the clock easier to use, especially when a quote is quite long. The problem is that in books, an impressive variation of time descriptions is used. It can be anything from '6.00 p.m.' or '18:11:00' to '0600h', 'around six o'clock', just 'at six', or 'twenty-eight minutes past eleven'. I made a script to try and find most of these variations, did the ones it couldn't find myself, and added them to the csv file.

If you want to make your own Kindle clock, you may use my scripts (find them attached below), but you can also just download all the resulting images.

https://www.instructabl... Download

https://www.instructabl... Download


https://www.instructabl... Download


https://www.instructabl... Download

Step 4: Starting and Stopping the Clock

I wanted to be able to start my literary clock by pressing the shortcut Shift+C on the small keyboard of the e-reader. Pressing it again stops the clock and turns the clock into a normal e-reader again.

First, create this folder: /mnt/us/timelit and then put the scripts I attached below in there.

The images (see previous step) go into /mnt/us/timelit/images and /mnt/us/timelit/images/metadata/

When you install the Launchpad hack, the folder /mnt/us/launchpad is created. Create a new file there called startClock.ini and put this text in there:

```
[Actions]
C = !sh /mnt/us/timelit/startstopClock.sh &
```

That creates the shortcut Shift+C. If we press that, the bash-script startstopClock.sh starts. It stops the Kindle framework (the normal user interface), prevents the Kindle from going into power save mode and creates a small file (/mnt/us/timelit/clockisticking) to indicate the clock has started.

Note: Shift+C on the Kindle is really 'press shift, let go, press c'.

If the user presses Shift+C again and the clockisticking file is already there, startstopClock.sh will remove it and restart the Kindle.

startstopClock.sh also executes another script, showMetadata.sh, to enable the keystrokes that will show the metadata (using the command /usr/bin/waitforkey). If the user pushes the 'next page' button on the sides of the Kindle, it will check if the clock is ticking and if it is, will show the same image as currently is shown (which file that is, is saved in the clockisticking file) but then with title and author at the bottom.

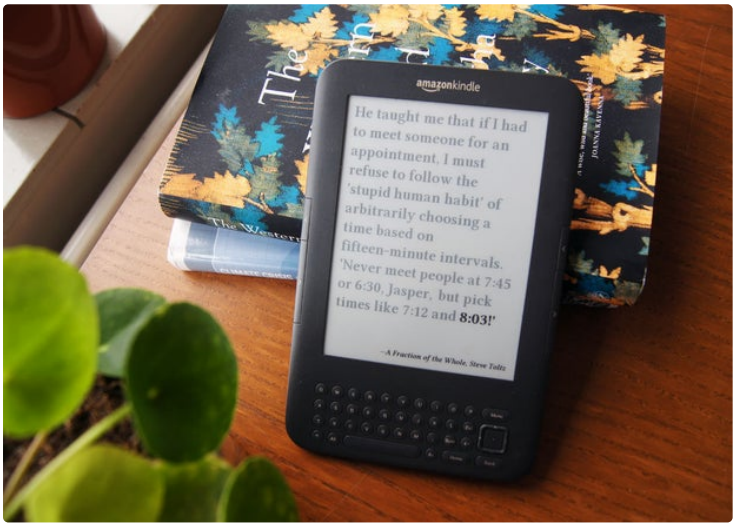
Changing the time on the display every minute is done by adding this line to /etc/crontab/root:

```
* * * * * sh /mnt/us/timelit/timelit.sh
```

and then restart crontab like this: /etc/init.d/cron restart

Every time it is run, timelit.sh checks if the 'clockisticking' file is created. If it is, timelit.sh proceeds to show the image for the current minute.

Note: you'll probably want to change the timezone in timelit.sh where it says 'TZ=CEST'.



Step 5: Making a Stand

I was inspired by other Instructables to make a concrete stand for my Kindle clock. I could also have made something out of wood (or even a book), but I liked to try cement because I never did before and also because I thought the grey color would go nicely with the e-reader.

I cut a piece of styrofoam the size of the e-reader, plus a little extra for the USB cable to go in. I wrapped it in cling film and a bit of clear tape, so the cement would come off easily afterwards. I taped it to the bottom of the food container using double-sided tape.

Then I mixed enough cement to fill the food container to about 5 centimeters (2") deep. I'm not sure, but I may not have used enough water, because the cement was less pourable than I had expected. I definitely should follow the concrete class before my next try :-)

I put the cement in the container using a garden shovel, tamped it a bit, and then let it dry for two days.

The next time I will try for a smoother surface by first sifting the cement to get rid of the small rocks, adding a bit more water and spend more time sanding the result. Then I will also make a small recess in the base so the USB cable goes to the back of the stand. This can be done using a straw.

Step 6: Further Ideas

The literary clock looks really nice, and the quiz part works well. My girlfriend now and then checks to see from which book a quotation is from (she usually guesses correctly :). The stand did not come out quite how I hoped, but

I'm looking forward to trying making a better one.

I will probably also add a lamp, either clamped on the device or incorporated into the new base. When the clock sits in the cupboard, sometimes it is a little too dark to be able to tell the time.

Instead of getting power for a lamp separately, one could power a lamp using power from the hinge slot in the Kindle. Two slots are there for Kindle cases that have a lamp built in. You'd have to open the Kindle and do some soldering, or make your own metal clamps, but that would be sweet. One could even connect a light sensor, so the lamp will only switch on when it's getting dark.

Extra features I hope to get round to are

- having the clock stop between 1am and 6am, to save power
- turn of wifi for the same reason, but turning it on daily for a couple of minutes to synchronise the system clock
- showing the percentage of the current minute that has passed as small blocks at the bottom, just like the Kindle indicates the progress the reader is making in a book
- show a warning when the Kindle's battery is running out

(these last two could be done by overlaying small images on the larger image using the Kindle's eips command, see my scripts for examples).

Other possible ideas are

- using keys on the Kindle to set the time
- show a default image when the clock starts and/or when no image is found
- using a shortcut (shift-Q for instance) to toggle quiz mode
- have the Big Ben sound chime at the top of the hour (only during the day), as the Kindle has a nice speaker built in. Other sounds could be the sound of slamming a book shut or turning pages or even reading out a quote.

I hope you like the idea and this Instructable. Let me know if you have any questions or suggestions!

Hello guys! I have a kindle touc! I think I have done everything right, but when I run the startStop script from PuTTY I get this errors: /mnt/us/timelit/startstopClock.sh: line 37: /etc/init.d/framework:

not found
/mnt/us/timelit/startstopClock.sh: line 37: /etc/init.d/powerd: not found

Any ideas on how to fix this? THANKS SO MUCH!

I really like the idea, so I build a web version using your quote list.

http://jenevoldsen.com/literature-clock/

awesome

I cannot find many details on how to get this working on a K5 Touch :(Can anyone assist?

Made with a Nook Simple Touch and housed in a hollowed-out old book. Since the Nook doesn't have the tool eips that Kindle does to work the display, I had to write a shellscript that's triggered every minute by a cron job and replaces the screensaver with an image corresponding to that minute. Then it quickly wakes and sleeps the Nook to retrigger the screensaver. A little hacky, but it works great and doesn't use much battery! The power cord is routed though the back of the book. I may write an instructable about how to do this with a Nook in the future and post my code.

Just e-reader clock ;)

Hi,

Thank you for the great tutorial and concept! I'd really like to surprise my wife with this.

I'm trying to get this to work on my Kindle WiFi 3G (B006) version. I've restored my unit to factory default 3 times and followed the instructions very closely including watching the YouTube video. But when I press Shift and then C, the only thing that happens on the screen is an indication on the bottom left that shows (^[C]) followed by the word, "Success!," however nothing else happens. The display continues to show the normal Kindle page, (dictionaries and archived items).

What could I possibly be missing? Thank you again for the awesome idea and any help you could provide!

Edited: I forgot to mention that everything including the jailbreak that I used were the k3g based on the Kindle I have. Maybe I'll try using the k3w versions and see if that works.

Did you ever find a solution to this? I'm also getting "Success!" then nothing.

Did you have any luck with this?

I'm so sorry to be late in responding. I never was able to figure this issue out. I will have time this coming week and am going to try the adjustments that TonyM311 spoke of. Here's hoping!

No worries! I actually got it to work shortly after posting! Thanks

That's awesome! May I ask what you changed to get it to work? I just tried it again and ended up with the same result.

I did this project a year ago now so I don’t remember off the top of my head. I don’t have the kindle,

I gave it as a gift but I'll be getting back into it in a few weeks to try and fix the daylight savings causing the clock to be wrong for half the year. When I'm looking at it I'll try and remember. I believe it was something similar to what TonyM311 suggested in a reply to my comment

Awesome! I was thinking that the changes that TonyM311 made might work. I'll be trying this very soon and will post my success...or my failure. Lol! Either way, I'll be learning more so it's a win win.

I am also in the exact same boat! Did you manage to find a way to fix it?

Did you have any luck with this?

Hey, my K3 was also booting to a blank screen. I figured out it was because I did not place the "litclock_annotated.csv" under the /mnt/us/timlit folder.

As soon as I place that in it booted up.

We're you able to figure out what was wrong with it?

I love your idea. I don't have a kindle, but your spreadsheet and images could be easily used to make a processing Android App / Java App / webpage. I will try to do it —though I guess someone has done it previously.

Thanks for this project, I had a lot of fun doing it!

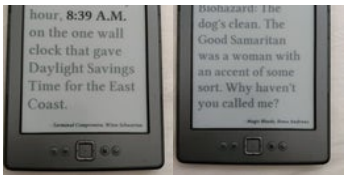
I've made some modifications to the original image generation script, so that it checks to see if the image already exists before generating a new one (which is handy if you just want to run the script to add one or two more images, since it only does the memory intensive stuff if the image isn't already there), and have uploaded that modified script to github.

https://github.com/zenbuffy/LiteraryClock

I've also added some new times to the csv file (also in github) and some new images (also in github) if anyone wants to grab some additional minutes for their clock.

Further plans include tweaking the setup a little since my kindle (and others it seems) doesn't have buttons, so the launch script idea doesn't quite work for me.

Thanks for the inspiration!



Awesome, thanks!

Hi, from the images on your post it looks like you have this running on a Kindle 4 (non-touch)??

I have several K4NTs and have tried to install the Literary Clock however when i start it running using the command line over SSH -

sh /mnt/us/timelit/startstopClock.sh

..the Kindle goes into a reboot loop and then shows the error "Your Kindle Needs Repair" and I have to use Kubrick to unbrick it, before jailbreaking and installing all the other necessary things again.

Any help is appreciated!!

(If i get it working I am then looking to change the time images to a series of comic images as i would like the Kindle to display a randomly selected comic image twice per day instead)

This fantastic idea inspired me to try it out with my old Kindle (non-touch version 4).
I hit a couple of road-bumps on the way, but nothing major - a good learning experience!
But, I hit one major roadblock. I could not get the Python installation to succeed. I thought all was lost until I realised that the python was only used to generate a random number to select between several quotes.
I know a little bit of shell scripting and thought there should be a way to generate a random integer without using python. As the Kindle uses a busybox shell which has some features removed (for instance there is no $RANDOM variable), this is a little tricky.
In the timelit.sh script you can replace this line:
ThisMinuteImage=$(find /mnt/us/timelit/images/quote_$MinuteOTheDay* 2>/dev/null | python -c "import sys; import random; print(''.join(random.sample(sys.stdin.readlines(), int(sys.argv[1])))).rstrip())" 1)

with these several lines.
#generate random number (compatible with busybox)
n=65536
while [$n -ge 65536]; do
n=1$(</dev/urandom tr -dc 0-9 | dd bs=5 count=1 2>/dev/null)
n=$((n-100000))
done
IMAGE_NUMBER=$n
let "IMAGE_NUMBER %= $lines"
ThisMinuteImage=$(find /mnt/us/timelit/images/quote_${MinuteOTheDay}_${IMAGE_NUMBER}.png)

Now, there is no need for the python installation.

Cool! I think I used Python because I could not find another way to get a random number. Thanks for this!

I have several K4NTs and have tried to install the Literary Clock however when i start it running using the command line over SSH -

sh /mnt/us/timelit/startstopClock.sh

..the Kindle goes into a reboot loop and then shows the error "Your Kindle Needs Repair" and I have to use Kubrick to unbrick it, before jailbreaking and installing all the other necessary things again.

Any help is appreciated!!

How did you work around the fact that there is no Launchpad for K4 NT?

Tried to install on kindle 4 but stuck because there is no launchpad for k4...

Thank your for the great tutorial. I finally made it work on a paperwhite. I started the startstopClock.sh via ssh and it worked. I rewrote the timelit.sh with a loop command, but as soon as i unplug the usb cable from my computer, the clock does not refresh. Any recommendations?
Thank you.

Thank you for your great tutorial. Can you maybe please add a description how i get it work on the paperwhite?
Thank you.

This opens up loads of ideas for an image based clock of any subject. Wouldn't it be great to collect a load of images from movies that show the time. You could slowly build up the time by just collecting 10min interval images from 1 to 12 (72 images) or 5min intervals (144) and expand on that. As long as you name the images correctly then it doesn't matter if you miss out images as the clock will only update if it has an image of the time.

I made this on PW1_3G, which I downgraded to 5.3.4 before jailbreaking. It was a tough project as I am not very literated in scripts or ssh. Still working on authors to show. But youtube videos by "1nfornica" helped me a lot for starting scripts.(https://www.youtube.com/watch?v=HflHaklYDWg&feature=youtu.be&t=225). Also, tips by Joe208 was helpful for PW1. Yet, I could have my kindle working by modifying cron along the Youtube than writing a script.

I did it and it works like a charm :

//www.youtube.com/embed/oNvO-eTQWTc !

Made some minor changes: I did not want to have the quizz part (I'm not english native speaker and my knowledge in english litterature is ... poor).

So, I update the script to directy display the "metadata" version :)

As read in another comment the shortcut key is really "SHIFT during few seconds and then C" ! :)

To conclude and let my kids be able to read this clock too, I started a french version of the text. I just started to gather french book extracts.

If you are french, or read french books, and want to participate, feel free to contact me ! :)))

For sure, I'll share back the result to the french community ;-)

This is such a great project. Thank you for sharing! I surprised myself in being able to get it running. I just have a couple issues:

1. It doesn’t necessarily update every minute. Sometimes it takes a couple of minutes off, then switches to a new image.

2. After about 10 minutes the Kindle lock screen comes up. I read above that this shouldn’t be happening.

Thoughts? Thanks again.

Not every time around the clock has an associated quote. If there is no quote, then there is no update.

It took some elbow grease to get this working on an old Kindle Paperwhite I had laying around but the result is really cool! Thanks for the inspiration and code! For anyone with a Paperwhite/Touch these are my high-level changes.

I couldn't get the script to control power, so I ran the ~ds command in the search bar to disable the screensaver.

You'll need a different javascript package too. The one that worked best for me was from this thread at MobileRead (showthread.php?t=277687).

I couldn't edit the cron root file to get this script to run at startup but I found a cool package called eventHandler on the MobileRead forums (showthread.php?t=198484) which lets you easily run a script at boot by putting in /mnt/us/scripts/startup.

Because I couldn't edit the cron root file I also couldn't get the timelit script to run every minute. My answer was to add some code to it (a while : style loop with a 60 second rest) so it updates once per minute.

Finally, I had the most frustrating time trying to get the timezone in this script working for me here in Denver instead of the default UTC. Turns out there's a bunch of great time zone data on the Kindle in /usr/share/zoneinfo which you can use to get the exact verbiage it's looking for.

For looks I went with a cheap wooden stand and a wood grain skin for the Kindle, I think it turned out quite nice! There are a few things left on the to-do list like using the PHP script to make pictures for the resolution of this device.

Thanks for these tips! I'm really close to having it working on a Kindle Touch 1.

Since the script power commands didn't work, I'm guessing you also disabled the '/etc/init.d/framework stop' command. How did you handle stopping the framework from refreshing? Does the eventHandler take care of that?

Thanks!

How did you get it to display the author and the book title at the bottom? Thanks!

There are two sets of photos in the instructions. One set is just the quote, and the other includes the author and book title. I think the intent was to display that extra info when you press a button but I just liked it more as the default option. In the script, there's a line with a reference to the pictures folder, just make sure it's pointing to the one with the author and book titles.

Similar to the web versions that people have created, I've put together a macOS screensaver using the same quotes and concept: https://github.com/mmattozzi/LiteraryClockScreenSaver ... enjoy!

I was unable to find a key code list for the K4 such that one could start-stop the script on command, so I am using Kite and running the startstopclock.sh as an "onboot" init-script.

*edited because the script may have caused a crash today, I will update.

Hey, did you get it working? I have a K4 also and don't know how to get the app to run. I don't use the kindle for reading at all, so I'd be fine with it being a full-time clock.

Yes! Well, partially, and good enough for me. I did the jailbreak, usbnet, python, cron updates, added KUAL for ease of USBnet toggling, and the Kindle actually just runs the timelit scripts on startup! The one issue is that the power daemon isn't affected (on my kindle) by the timelit scripts, so I disabled it in the shell by typing "lipc-set-prop -- com.lab126.powerd preventScreenSaver 1", since my keyboard commands wouldn't disable the screensaver either. I removed Kite and batch renamed all the metadata image files so they would display as it runs(didn't care about the quiz functionality), and so far it's been running for a week on the same charge as it started with.

I did modify timelit.sh to remove the time zone after I registered the kindle and have just been using system time and it works quite well!

Hi, can you please check your links, i cannot download the images.zip file... comes back with a 503 error....

Thanks, cant wait to get a start on this, seems link an awesome use of an old kindle..

It works for me. Probably a temporary error.

My apologies, should also say thank you for posting the instructable...

Hi, Thanks for the prompt response. Looks like it is an issue with one of my laptops, downloaded the file from a tablet...

Awesome project... Got it working on a Kindle Touch..

Hello, have anyone please experience with battery life?

I'm experiencing that battery in Kindle 3 with disabled wifi last only for 10 or 20 hours (not sure).

If your battery is old it might not hold its charge as long, maybe you should buy a replacement.

Thanks for reply. Did you test how battery last for you?

On my old kindle, the battery lasts a couple of days