

TinyGo, Petit mais costaud



Aurélie Vache - [@aurelievache](https://twitter.com/aurelievache)



Thierry Chantier - [@TitiMoby](https://twitter.com/TitiMoby)



Un Grand MERCI à nos sponsors 2025



CGI



open

MOODY'S

WISEO
— POSITIVE DIGITAL MAKERS —



CRITEO



elastic



clever cloud



zenika



bpifrance.i0

kellogggroup

Aurélie Vache

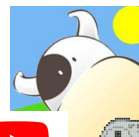
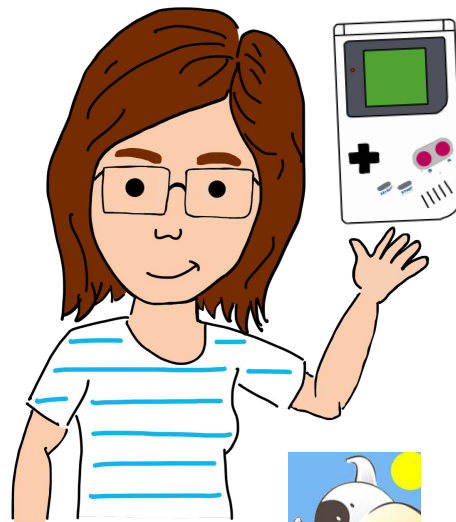
 @aurelievache

Developer Advocate chez  OVHcloud

Organisatrice de conférences

Rédactrice d'articles & de livres tech

Sketchnoteuse ... &  Retrogaming



Les Productions de MOA



<https://developers.events>



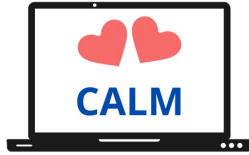
<https://www.youtube.com/AurelieVache>



<https://dev.to/aurelievache/>



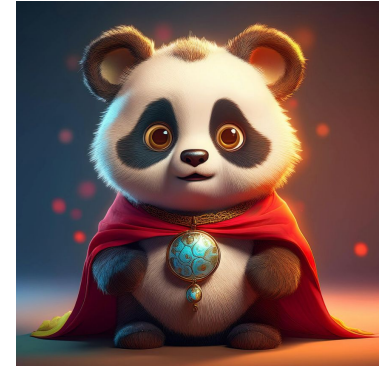
Thierry Chantier



 GDG Cloud Lyon



Developer Advocate
OVHcloud



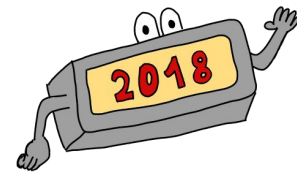
 TitiMoby@mamot.fr

 TitiMoby

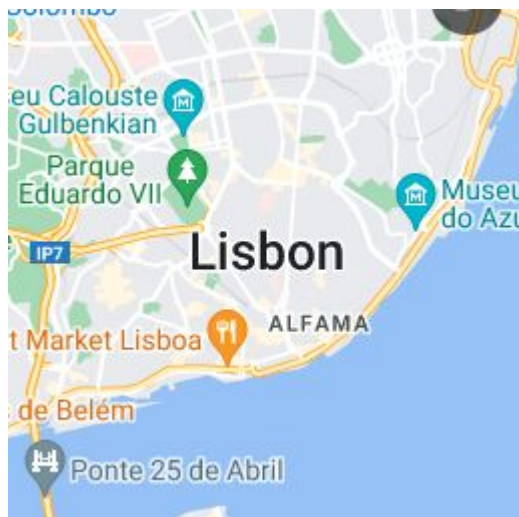
 <https://noti.st/titimoby>



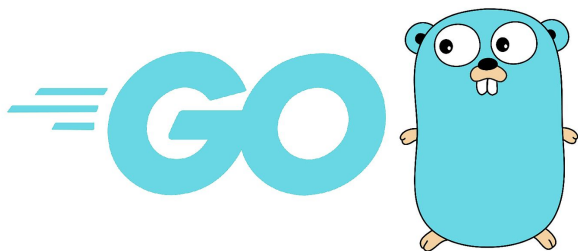
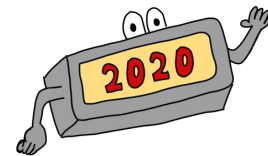
La petite histoire 🐼





 Google Developer Groups



La petite histoire



```
Tonton Codeur - Cahier de vacances - arguments  
```

réhabitue à cette façon de penser des années 70 et du C)

L'appel se fait alors de cette façon, en se rappelant que les arguments sont récupérés par des pointeurs.

```
response_err := http.Get(urlPtr * * &appId* * * apiKeyPtr * * &units
=metric *)
```

Pour lancer doobby, on peut continuer en mode 'go run'

```
~/Dobby/src$ go run doobby.go -url=http://api.openweathermap.org/data/2.
5/weather?q=Chamonix -apiKey=votre Api Key
```

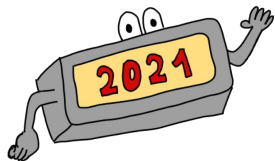
Me voilà équipé d'un système d'arguments, qui fournit également des services comme une aide en ligne avec l'argument '-h'

```
~/Dobby/src$ go run doobby.go -h
```

Usage of /tmp/go-build883761866/b801/exe/doobby:
-apiKey string
ApiKey a utiliser pour appeler votre url.
-url string
Url que Dobby va appeler pour vous.
exit status 2



La petite histoire



Tinygo

A collection of 5 posts

TINYGO

TinyGo - Joystick et boutons

Toute cette série est bien jolie, mais une console de jeu sans moyens d'interaction, c'est pas génial. Je vous propose aujourd'hui de voir comment contrôler le joystick et les boutons de la Pygamer. Je rappelle ici le disclaimer de l'article sur les fontes : la

 **THIERRY**
27 AOÛT 2021 · 3 MIN READ

TINYGO

TinyGo - Les fontes

Utiliser les Neopixels de la Pygamer peut déjà apporter un moyen de communication visuelle mais il faut pouvoir écrire des textes sur l'écran. Pour cela, TinyGo propose un module appelé TinyFont. Ce projet fourni même un outil pour générer vos fontes au format utilisable

 **THIERRY**
23 AOÛT 2021 · 3 MIN READ

TINYGO

TinyGo - Show time !

Après les deux premiers articles, vous êtes maintenant capables de manipuler l'écran et la led arrière de la Pygamer. Mais est ce que ce ne serait pas mieux de pouvoir faire un petit show lumineux avec les 5 Neopixels en façade ? Voyons voir ce

 **THIERRY**
20 AOÛT 2021 · 2 MIN READ

TINYGO

TinyGo - Allumons l'écran de la Pygamer

Maintenant que nous avons vu comment mettre en place un environnement qui permet de coder avec TinyGo pour la Pygamer, voyons ce que l'on peut faire de l'écran de cette console. Vous l'avez peut être vu dans la documentation d'installation, mais il existe des

 **THIERRY**
18 AOÛT 2021 · 2 MIN READ

TINYGO

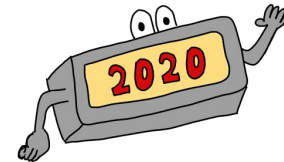
TinyGo - et si on codait une Pygamer

Tout comme je l'avais fait l'année dernière, je suis parti en vacances avec mon iPad mais cette fois avec quelques idées de code autour du langage Rust. J'ai appris pas mal de choses avec ce langage mais il me manquait cette petite touche de

 **THIERRY**
15 AOÛT 2021 · 3 MIN READ



La petite histoire 🐮 - Wow!!!



FOSDEM 2020



conejo 🐰🐇🐇 @conejo@social.tinygo.org @_CONEJC · 12 janv. 2020 ...

The future is this: #GBA games made in #Golang 🐼 This and a few other toys at my #FOSDEM talk about @TinyGolang >> [fosdem.org/2020/schedule/...](https://fosdem.org/2020/schedule/)



0:00



8



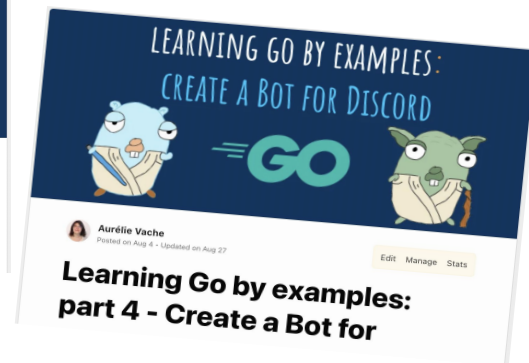
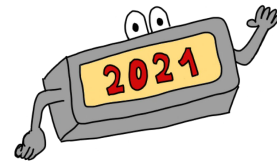
62



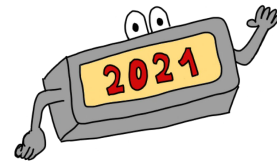
181



La petite histoire 🐮 - série d'Article sur Go



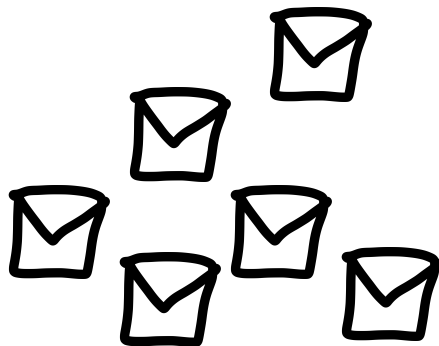
La petite histoire 🐰 - Expérimentations TinyGo



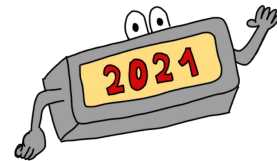
@conejo

```
● ● ●  
$ tinygo build -target=gameboy-advance -o bin/gba-display.gba  
gba-display.go ; mgba bin/gba-display.gba  
tinygo:ld.lld: warning: lld uses blx instruction, no object  
with architecture supporting feature detected  
The game crashed!
```

```
● ● ●  
$ tinygo run -target=gameboy-advance  
gba-display.go  
tinygo:ld.lld: warning: lld uses blx  
instruction, no object with  
architecture supporting feature  
detected
```



La petite histoire 🐮 - série d'Article sur Go



LEARNING GO BY EXAMPLES:
CREATE AN HTTP REST API SERVER

Aurélié Vache
Posted on Jul 21 - Updated on Aug 27

Learning Go by examples:
part 2 - Create an HTTP REST
API Server in Go

Edit Manage Stats

LEARNING GO BY EXAMPLES:
CREATE A CLI APPLICATION

Aurélié Vache
Posted on Jul 27 - Updated on Aug 27

Learning Go by examples:
part 3 - Create a CLI app in Go

Edit Manage Stats

LEARNING GO BY EXAMPLES:
CREATE A BOT FOR DISCORD

Aurélié Vache
Posted on Aug 4 - Updated on Aug 27

Learning Go by examples:
part 4 - Create a Bot for

Edit Manage Stats

LEARNING GO BY EXAMPLES:
CREATE A GUI/DESKTOP & MOBILE APP

Aurélié Vache
Posted on Aug 22 - Updated on Aug 27

Learning Go by examples:
part 7 - Create a cross-
platform GUI/Desktop app in
Go

Edit Manage Stats

LEARNING GO BY EXAMPLES:
CREATE A GAME BOY ADVANCE GAME

Aurélié Vache
Posted on Aug 11 - Updated on Aug 27

Learning Go by examples:
part 5 - Create a Game Boy
Advance (GBA) game in Go

Edit Manage Stats

LEARNING GO BY EXAMPLES:
CREATE A GRPC APP

Aurélié Vache
Posted on Aug 18 - Updated on Aug 27

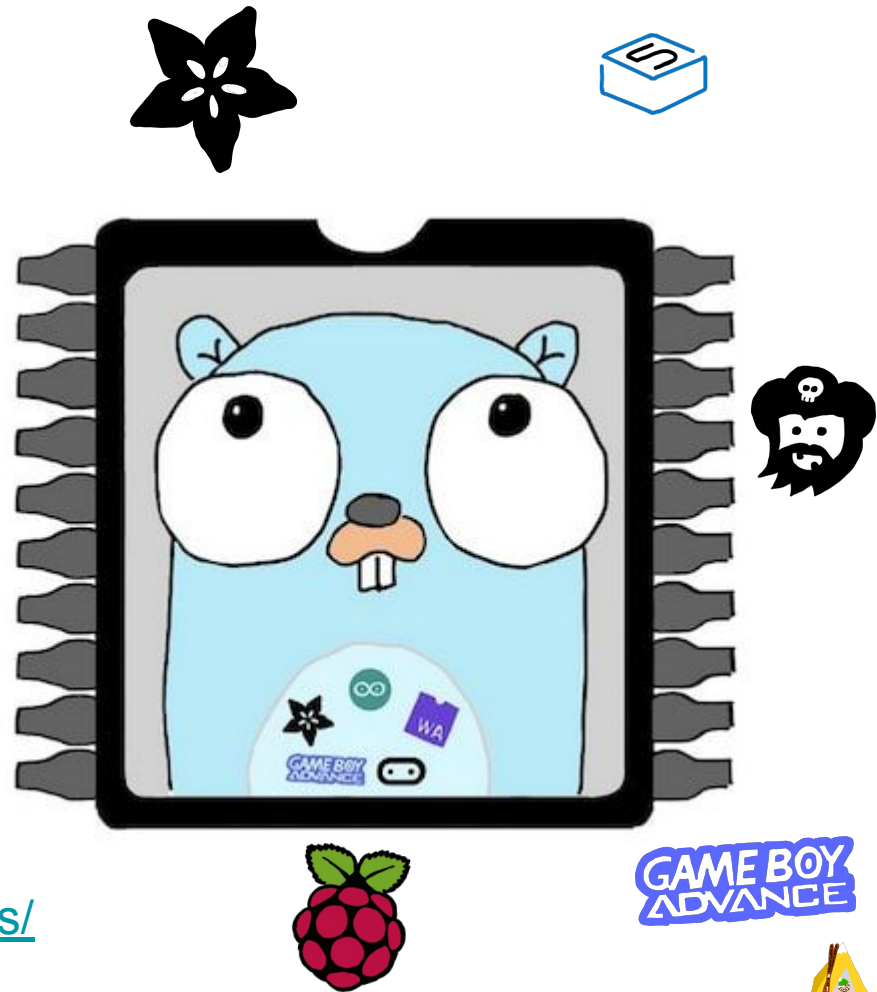
Learning Go by examples:
part 6 - Create a gRPC app in
Go

Edit Manage Stats



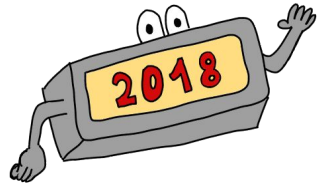
“Go compiler for small places” :

- Microcontrollers
- Web Assembly





TINYGO

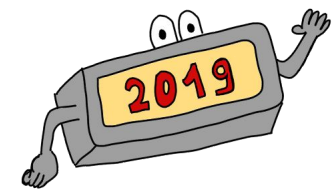


7/06/2018 : Le tout premier commit grâce à Ayke van Laethem, TinyGo est né ! 🎉



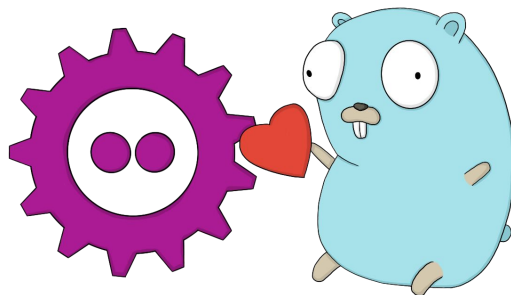


TINYGO



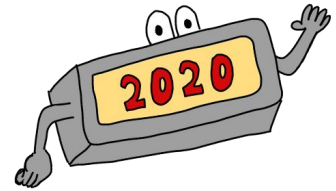
01/02/2019 : Première release v0.1.

03/02/2019 : Au FOSDEM TinyGo est mis en avant & l'évangélisation est alors portée par Ron Evans (@deadprogram).





TINYGO



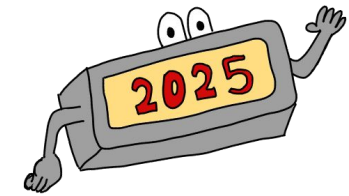
02/02/2020 : Toujours très visible au FOSDEM, TinyGo devient un projet sponsorisé officiellement par Google.


L'arrivée de WASM en cible de compilation confirme le succès de TinyGo.





TINYGO

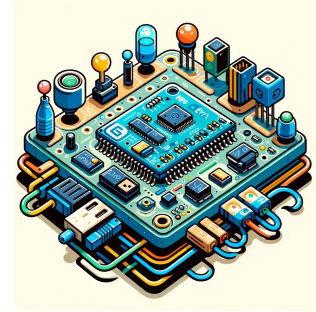


Le projet sur GitHub
a dépassé les 15K 





Go vs TINYGO



VS





Go vs TINYGO

```
go build -o hello-go main.go
```

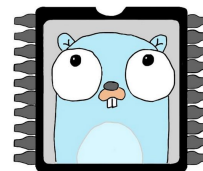
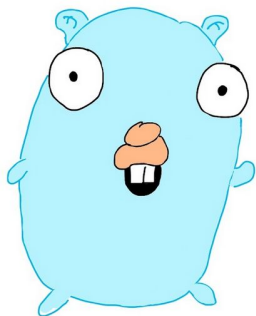
1,5M hello-go

|

```
tinygo build -o hello-tinygo
```

114K hello-tinygo

```
func main() {  
    println("hello TinyGo lovers")  
}
```





TINYGO: les petits secrets





TINYGO CLI: comment l'installer ?



```
// Sur MacOS
```

```
$ brew install tinygo-org/tools/tinygo
```

```
// Sur Linux (Debian)
```

```
$ wget https://github.com/tinygo-org/tinygo/releases/download/v0.35.0/tinygo_0.35.0_amd64.deb
```

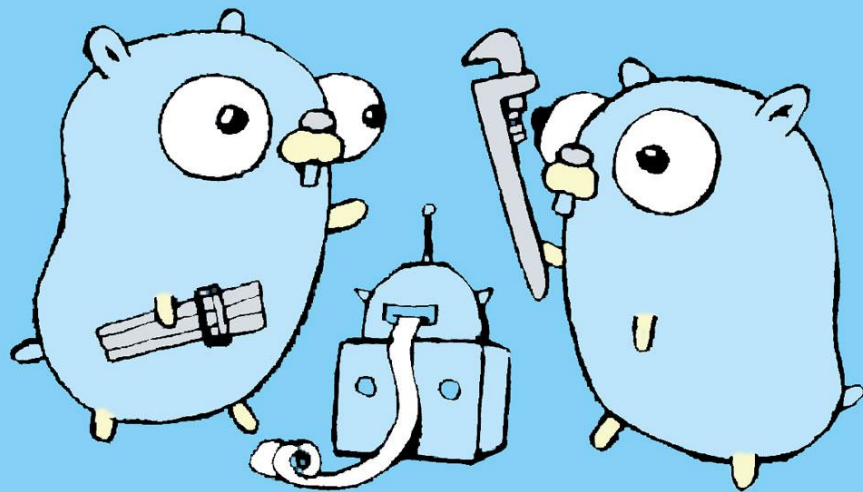
```
$ sudo dpkg -i tinygo_0.35.0_amd64.deb
```

```
// Sur Windows
```

```
$ scoop install tinygo
```

<https://tinygo.org/getting-started/install/>



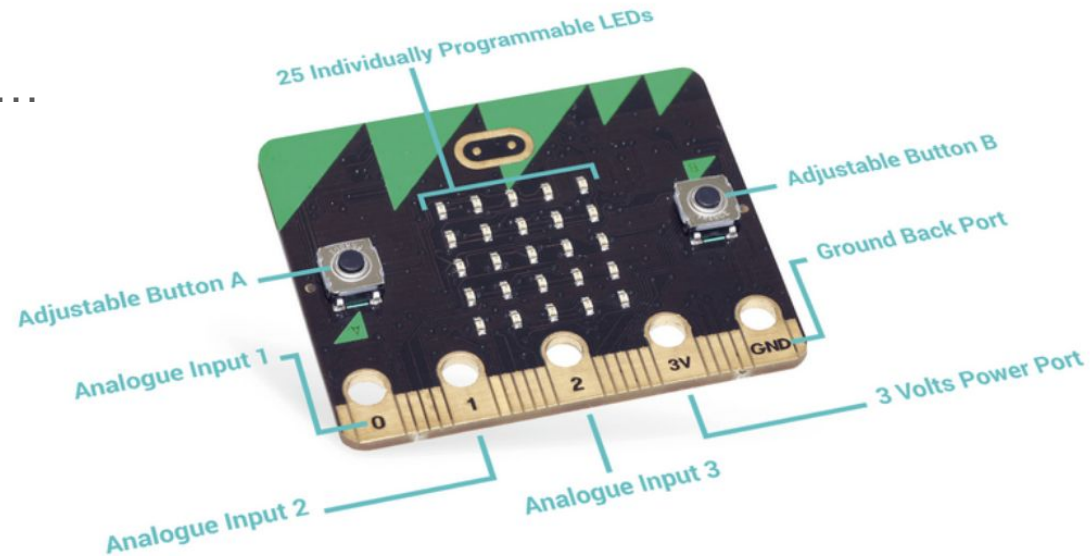


Démos !



Microbit

- Création en 2016 à l'initiative du gouvernement UK
- Simple mais complète
- Programmable par bloc, Python...



<https://microbit.org/>



GopherBadge

- Création en 2023 par Daniel “@conejo” Esteban
- Pensé pour promouvoir TinyGo
- Basé sur un Raspberry Pi RP2040
- 1 LED bleue
- 2 LEDs RGB Neopixel
- Accéléromètre LIS3DHTR
- Ecran 320x240 IPS TFT
- 6 boutons
- buzzer



<https://gopherbadge.com/>





Game Boy Advance





Game Boy Advance



- Ecran 240x160px
- 32 768 couleur (15-bit)
- Un pad multidirectionnel, boutons "A", "B", "L", "R", "START" & "SELECT"
- Core ARM7TDMI 32 bits avec mémoire intégrée





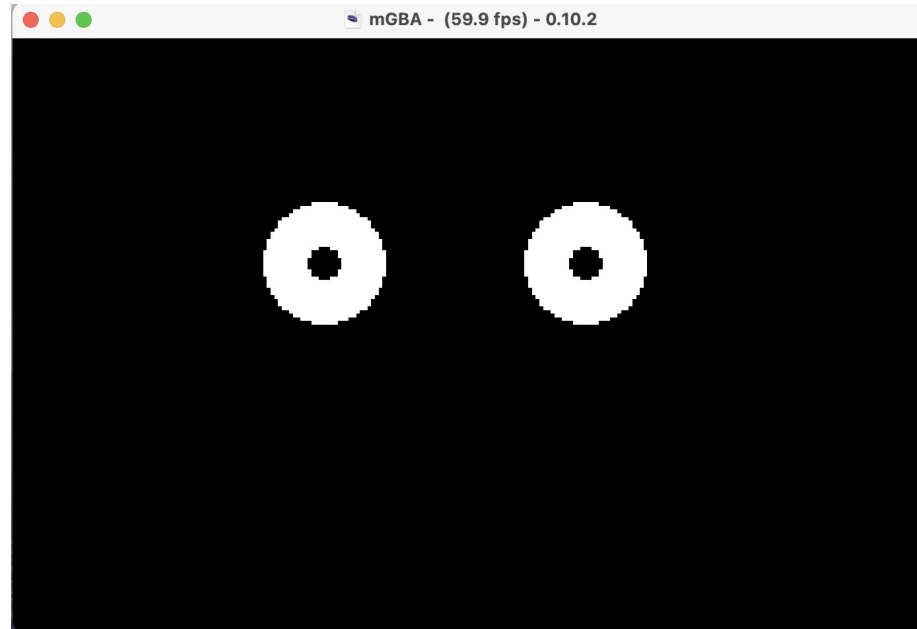
Game Boy Advance > demo





Game Boy Advance > demo

2,7K 3 jan 18:42 06-Eyes/eyes.gba



<https://github.com/titimoby/tinygo-examples/tree/main/gba/06-Eyes>





Game Boy Advance > demo





Game Boy Advance > demo

33K 4 jan 13:50 03-Gopher/gopher.gba



<https://github.com/titimoby/tinygo-examples/tree/main/gba/03-Gopher>





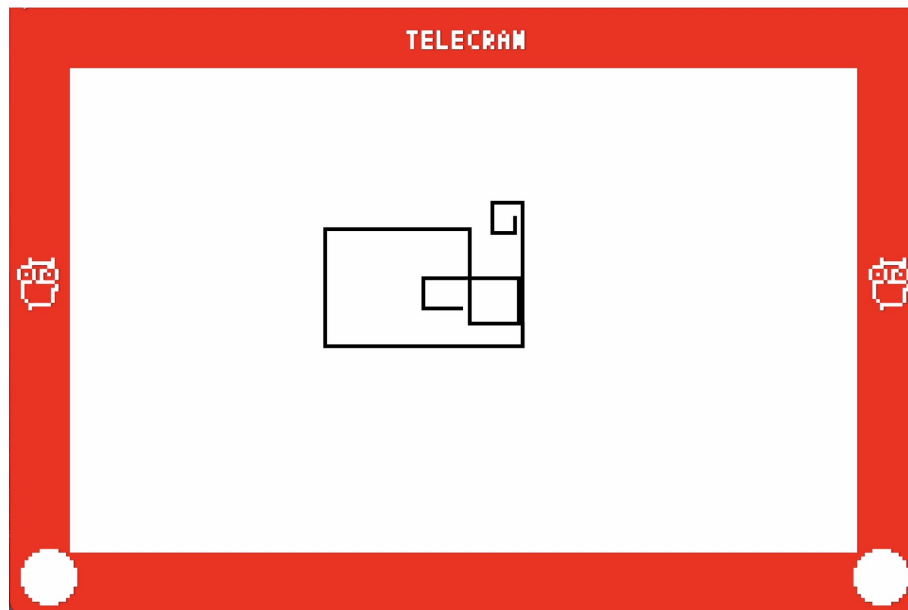
Game Boy Advance > demo





Game Boy Advance > demo

14K 3 jan 20:55 08-LetsDraw/telecran.gba



<https://github.com/titimoby/tinygo-examples/tree/main/gba/08-LetsDraw>





Game Boy Advance

Commande :



```
$ tinygo build -o telecran.gba -target=gameboy-advance main.go
```



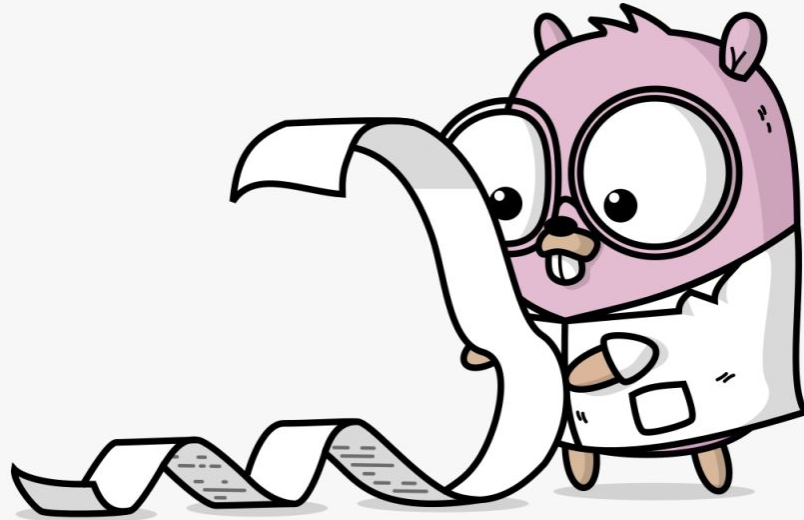
Emulateur :



<https://mgba.io/>



Outils pratiques





TinyDraw

Permet de dessiner des figures géométriques (basé sur la librairie [Adafruit GFX](https://www.adafruit.com/product/109)).

```
white = color.RGBA{255, 255, 255, 255}
green = color.RGBA{0, 255, 0, 255}
red   = color.RGBA{255, 0, 0, 255}

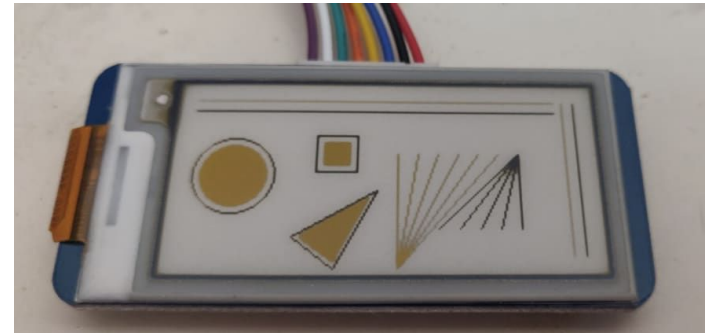
// ...

tinydraw.Line(&display, 100, 100, 40, 100, red)

tinydraw.Rectangle(&display, 30, 106, 120, 20, white)
tinydraw.FilledRectangle(&display, 34, 110, 112, 12, green)

tinydraw.Circle(&display, 120, 30, 20, white)
tinydraw.FilledCircle(&display, 120, 30, 16, red)

tinydraw.Triangle(&display, 120, 102, 100, 80, 152, 46, white)
tinydraw.FilledTriangle(&display, 120, 98, 104, 80, 144, 54, green)
```



<https://github.com/tinygo-org/tinydraw>





TinyFont

Permet d'afficher du texte/d'utiliser des "fonts" (basé sur la librairie [Adafruit GFX](https://github.com/adafruit/Adafruit_GFX)).

```
tinyfont.WriteLineRotated(&display,  
    &freemono.Bold9pt7b,  
    55, 60,  
    "@tinyGolang",  
    yellow,  
    tinyfont.ROTATION_90)  
  
tinyfont.WriteLineColorsRotated(&display,  
    &freemono.Bold9pt7b,  
    45, 180,  
    "tinyfont",  
    [color.RGBA{yellow, black},  
    tinyfont.ROTATION_270)
```



<https://github.com/tinygo-org/tinyfont>





TinyFont > 2ttf playground

Permet de tester des fonts avant de les utiliser.



<http://2ttf.com/typetest.php?id=HCQ3PvcaQ4U>





TinyGBA

Librairie pour TinyGo permettant de développer pour la Game Boy Advance.

```
var (
    //KeyCodes / Buttons
    keyDOWN    = uint16(895)
    keyUP      = uint16(959)
    keyLEFT    = uint16(991)
    keyRIGHT   = uint16(1007)
    keyLSHOULDER = uint16(511)
    keyRSHOULDER = uint16(767)
    keyA       = uint16(1022)
    keyB       = uint16(1021)
    keySTART   = uint16(1015)
    keySELECT  = uint16(1019)

    // Register keypad
    regKEYPAD = (*volatile.Register16)(unsafe.Pointer(uintptr(0x04000130)))
)

func update(interrupt.Interrupt) {
    // Read uint16 from register regKEYPAD that represents the state of current buttons pressed
    // and compares it against the defined values for each button on the Gameboy Advance
    switch keyValue := regKEYPAD.Get(); keyValue {

        case keySTART:
            // do what you want
        case keyRIGHT:
            // do what you want
    }
}
```

< Sans
tinygba



Avec
tinygba >

```
func update() {
    key := tinygba.ReadButtons()

    switch {
    case tinygba.ButtonStart.IsPushed(key):
        // do what you want
    case tinygba.ButtonRight.IsPushed(key):
        // do what you want
    }
}
```

<https://github.com/tinygo-org/tinygba>





TINYGO PLAYGROUND

TinyGo Playground

Console (TinyGo) * ▾

Flash

About

```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     fmt.Println("Hello chocolaines lovers!")
9 }
```

TERMINAL

Running...

Hello chocolaines lovers!

Exited

<https://play.tinygo.org/>





VsCode extension



TinyGo v0.4.0

TinyGo | ↻ 8 024 | ★★★★★ (2)

TinyGo support for Visual Studio Code

[Installer](#) ⚙️

[Détails](#) [Contributions](#) [Dépendances](#)

Visual Studio Code support for TinyGo

This is a simple extension to add TinyGo support to Visual Studio Code.

```
155 // enable enables all peripherals that might be disabled w
156 func enable() {
157     // Enable LEDs.
158     machine.SPI0.Configure(machine.SPIConfig{
159         Frequency: spiFrequency,
160         Mode:      0,
161         SCK:       spiClockPin,
162         SD0:       Set the pin to high or low.
163         SDI:       (machine.Pin).Set on pkg.go.dev
164     })
165     if mosfetPin != nil {
166         func (machine.Pin).Set(high bool)
```

Catégories

Programming Languages

Ressources d'extension

[Place de marché](#)
[Dépôt](#)
[Licence](#)

Plus d'informations





Ressources

- <https://github.com/titimoby/tinygo-examples/>
- <https://dev.to/aurelievache/learning-go-by-examples-part-5-create-a-game-boy-advance-gba-game-in-go-5944>
- <https://tontoncodeur.fr/tinygo/>
- <https://k33g.hashnode.dev/wasi-first-steps>



<https://gophers.slack.com/messages/CDJD3SUP6/>





Conclusion





TinyGo, un monde parfait ?

Débuter est un peu difficile :

- **Documentation en progrès mais pas encore parfaite**





TinyGo, un monde parfait ?

Débuter est un peu difficile :

- Documentation en progrès mais pas encore parfaite
- **Des exemples ... pas forcément mis à jour et clairs**

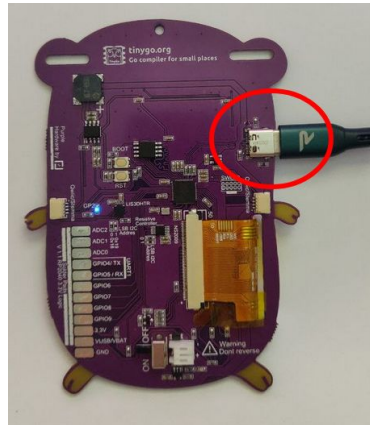




TinyGo, un monde parfait ?

Débuter est un peu difficile :

- Documentation en progrès mais pas encore parfaite
- Des exemples ... pas forcément mis à jour et clairs
- **Mais, des progrès ont été fait grâce à Gopher Badge et @conejo 🦾**



<https://github.com/conejoninja/gopherbadge/tree/main/tutorial/basics>





TinyGo, un monde parfait ?

Débuter est un peu difficile :

- Certains drivers sont encore en version “expérimentales” donc soumis à changements (GBA...)





TinyGo, un monde parfait ?

Débuter est un peu difficile :

- Certains drivers sont encore en version “expérimentales” donc soumis à changements (GBA...)
- **Certaines fonctionnalités sont commencées mais pas terminées**





TinyGo, un monde parfait ?

Pourquoi c'est tellement cool de se lancer :

- **Apprendre un nouveau langage**

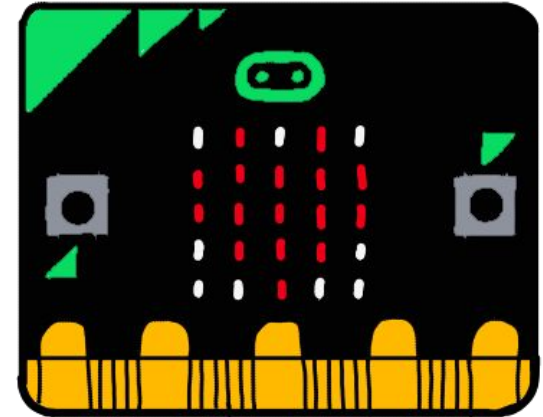




TinyGo, un monde parfait ?

Pourquoi c'est tellement cool de se lancer :

- Apprendre un nouveau langage
- **Bricoler et s'amuser avec de l'électronique**

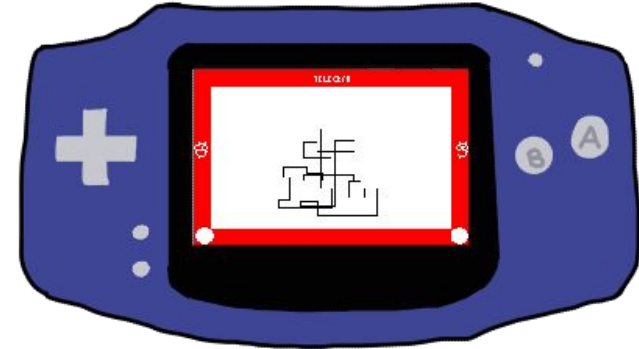




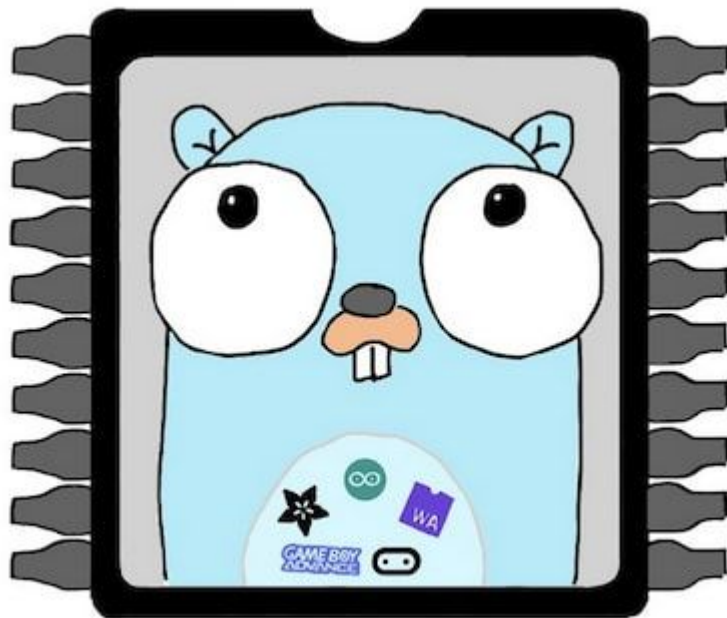
TinyGo, un monde parfait ?

Pourquoi c'est tellement cool de se lancer :

- Apprendre un nouveau langage
- Bricoler et s'amuser avec de l'électronique
- ... **Faire vivre sa GBA c'est tellement cool !**

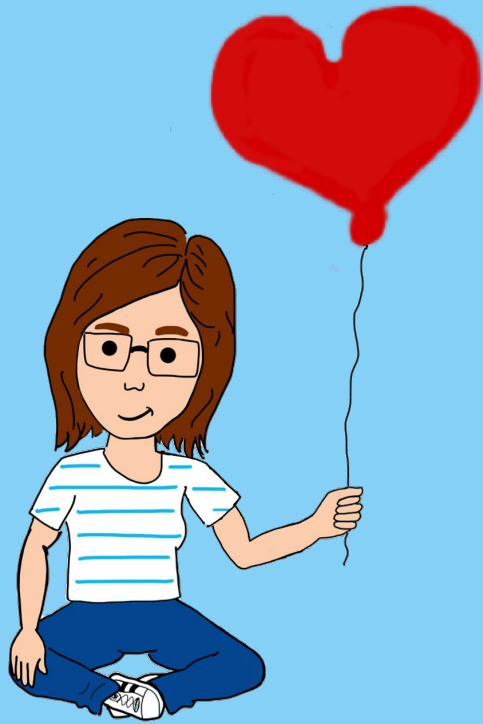


TinyGo, un monde parfait ?

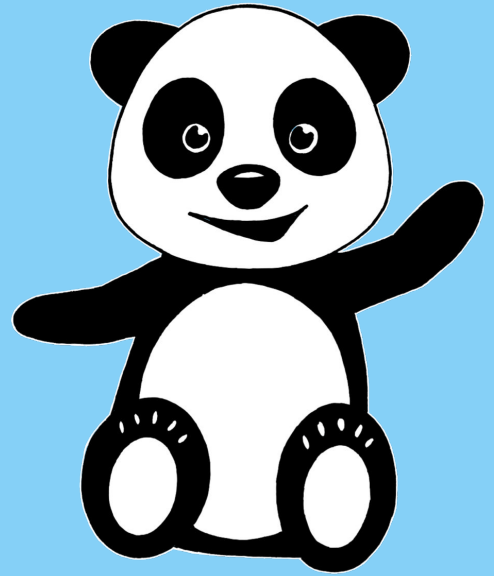


A vous de jouer !





MERCI



<https://ovh.to/FEMVA66>





QUESTIONS ?

