

Pushing (a little) the limits of your DIY electronics

Sanpi

30 juin 2016

Introduction

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

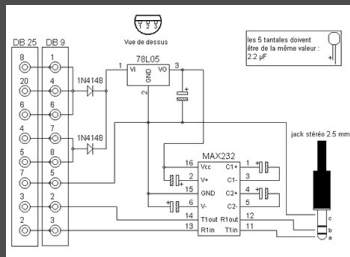
Arduino

Circuit imprimé

CMS

FPGA

Références



LE 21^e VIRUS INFO

Les IPods dangereux!

Père-Noël, la fin ?

Windows interdit en Grèce ?

Fabriquez votre console !

Nouvelles consoles

Les tests exclusifs de Shionos aux NDS, NDSi, NDSi-G, GBA, PSP, PDA, W5, Crystal...

Page 30, les liens Internet / Novembre 2003 / Vendredi / 15.00 FF / 1.026 FF / 8.979 F / 4.615 F / 1.440 CC / 180.000 F / 2.500 CC / 1.014 F / 101.14 F

ISSN 0947-3777 - 2006-10



Les VRAIS tests de toutes les consoles

PlayStation 2 - Xbox - GameCube - Game Boy Advance...
DYNAMIQUE

Fabriquez une (vraie) console de jeux !

De nombreux magasins vous ont déjà proposé de taper des listings de jeux vidéo. Tétris étant un exemple classique, qu'on ne présente plus. Cette fois, nous irons plus loin car nous vous proposons de réaliser votre propre console pour une poignée d'euros. Carrément !
Raft (gros merci à Richard Gurnée !)

Il se agit d'une console basée sur le processeur 8088 figurant sur les cartes de format 5.25 pouces. Ce processeur est capable de générer des données numériques à l'aide de sa sortie parallèle. Cette sortie, en parallèle avec les données numériques, est utilisée pour générer des données analogiques. Les données numériques sont envoyées dans un bus à 8 bits sur la sortie D.

Cette carte permet de générer des données numériques à l'aide de sa sortie parallèle. Cette sortie, en parallèle avec les données numériques, est utilisée pour générer des données analogiques. Les données numériques sont envoyées dans un bus à 8 bits sur la sortie D.



Fonctionnement de la console

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.



Figure 4 - Réglage de la console

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

le jeu !

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Comment on joue ?

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

le jeu !

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

le jeu !

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Comment on joue ?

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

le jeu !

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.



Figure 1 - Déclenchement de la console



Figure 2 - Réglage de la console de jeu

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Prix	
1. PCI 1000	1.000
2. PCI 1000	1.000
3. PCI 1000	1.000
4. PCI 1000	1.000
5. PCI 1000	1.000
6. PCI 1000	1.000
7. PCI 1000	1.000
8. PCI 1000	1.000
9. PCI 1000	1.000
10. PCI 1000	1.000
11. PCI 1000	1.000
12. PCI 1000	1.000
13. PCI 1000	1.000
14. PCI 1000	1.000
15. PCI 1000	1.000
16. PCI 1000	1.000
17. PCI 1000	1.000
18. PCI 1000	1.000
19. PCI 1000	1.000
20. PCI 1000	1.000

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Prix	
1. PCI 1000	1.000
2. PCI 1000	1.000
3. PCI 1000	1.000
4. PCI 1000	1.000
5. PCI 1000	1.000
6. PCI 1000	1.000
7. PCI 1000	1.000
8. PCI 1000	1.000
9. PCI 1000	1.000
10. PCI 1000	1.000
11. PCI 1000	1.000
12. PCI 1000	1.000
13. PCI 1000	1.000
14. PCI 1000	1.000
15. PCI 1000	1.000
16. PCI 1000	1.000
17. PCI 1000	1.000
18. PCI 1000	1.000
19. PCI 1000	1.000
20. PCI 1000	1.000

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko. Le jeu est chargé dans une mémoire de 16 Ko.

Pushing (a little) the limits of your DIY electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

ERGA

Recherche

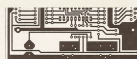


Figure 1 - Carte

Introduction

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

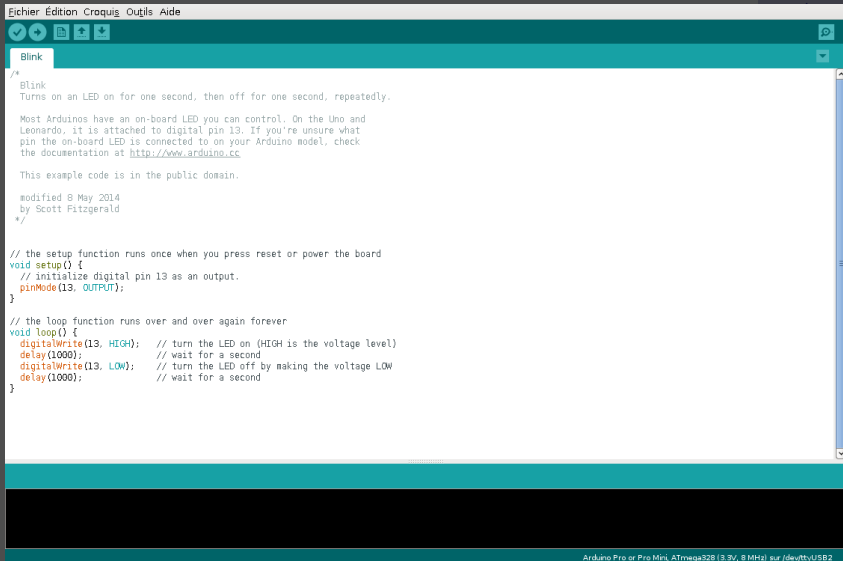
Arduino

Circuit imprimé

CMS

FPGA

Références



The screenshot shows the Arduino IDE interface. At the top, there is a menu bar with 'Eichier', 'Édition', 'Croquis', 'Outils', and 'Aide'. Below the menu bar is a toolbar with icons for file operations and a search icon. The main window title is 'Blink'. The code editor contains the following text:

```
/*  
Blink  
Turns on an LED on for one second, then off for one second, repeatedly.  
  
Most Arduinos have an on-board LED you can control. On the Uno and  
Leonardo, it is attached to digital pin 13. If you're unsure what  
pin the on-board LED is connected to on your Arduino model, check  
the documentation at http://www.arduino.cc  
  
This example code is in the public domain.  
  
modified 8 May 2014  
by Scott Fitzgerald  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin 13 as an output.  

```

At the bottom of the IDE, a status bar indicates: 'Arduino Pro or Pro Mini, ATmega328 (3.3V, 8 MHz) sur /dev/tty/USB2'.

```
BOARD_TAG=duemilanove
MONITOR_PORT=/dev/usb*
MCU=atmega328p
VARIANT=standard
F_CPU=16000000
AVRDUDE_ARD_BAUDRATE=57600
AVRDUDE_ARD_PROGRAMMER=arduino

include /usr/share/arduino/Arduino.mk
```

```
/usr/share/arduino/arduino-builder  
--hardware=/usr/share/arduino/hardware/  
--tools=/usr/share/arduino/tools  
--libraries=/usr/share/arduino/libraries/  
-fqbn=arduino:avr:uno  
sketch.ino
```

L'IDE

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le logiciel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le logiciel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

```
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

Le logiciel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

```
#include <avr/io.h>
#include <util/delay.h>

static void setup(void)
{
    DDRB |= (1 << PORTB5);
}

static void loop(void)
{
    PORTB |= (1 << PORTB5);
    _delay_ms(1000);
    PORTB &= ~(1 << PORTB5);
    _delay_ms(1000);
}

int main(void)
{
    setup();

    while (1) {
        loop();
    }

    return 1;
}
```

Le logiciel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

```
.org 0x0000
rjmp setup

setup:
    ldi r16, 0xFF
    out DDRB, r16
    rjmp loop

loop:
    sbi PORTB, 5
    rjmp delay
    cbi PORTB, 5
    rjmp delay
    rjmp loop

delay:
    in r17, TCNT0
    cpi r17, 255
    brsh dim
```

Le logiciel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Bootloader

Pushing (a little) the limits of your DIY electronics

Sanpi

Introduction

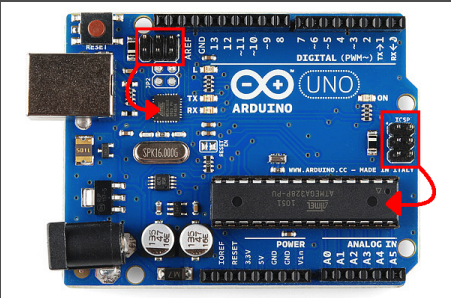
Arduino

Circuit imprimé

CMS

FPGA

Références



Bootloader

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références



Bootloader

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

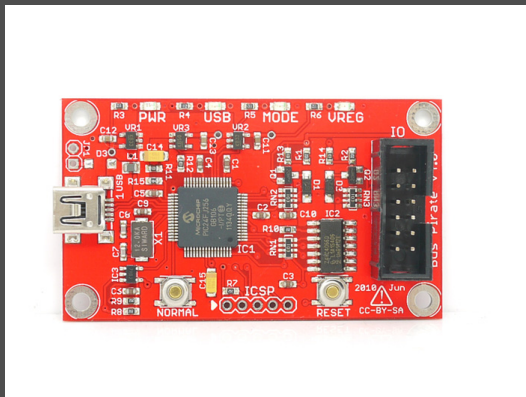
Arduino

Circuit imprimé

CMS

FPGA

Références



Bootloader

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

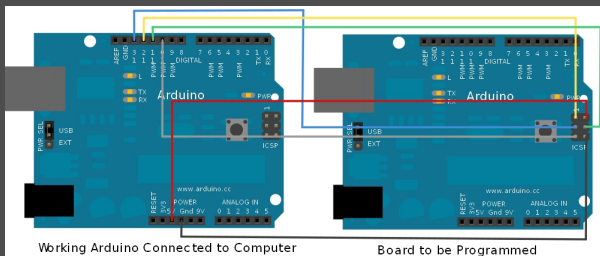
Arduino

Circuit imprimé

CMS

FPGA

Références



Bootloader

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

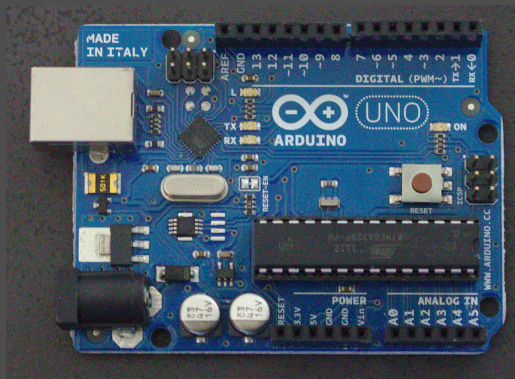
Arduino

Circuit imprimé

CMS

FPGA

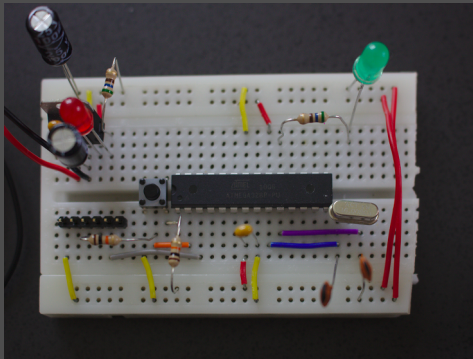
Références



Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi



Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

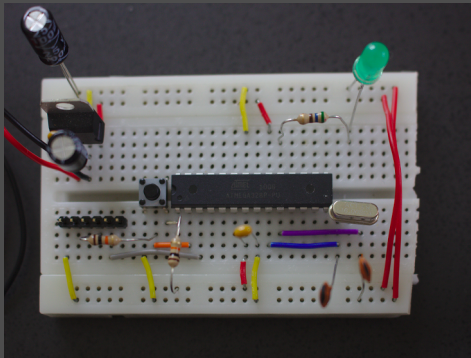
Arduino

Circuit imprimé

CMS

FPGA

Références



Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

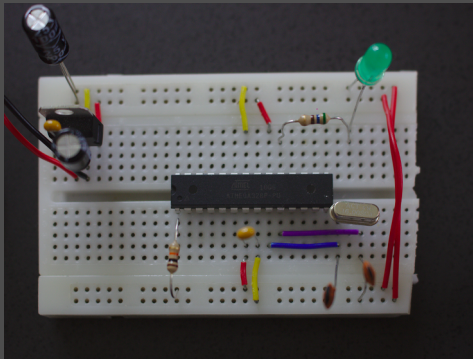
Arduino

Circuit imprimé

CMS

FPGA

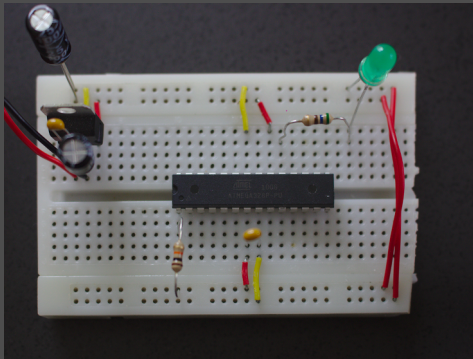
Références



Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi



Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

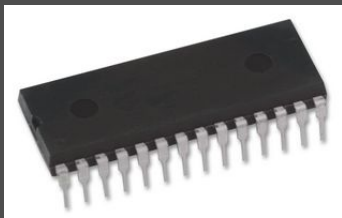
Arduino

Circuit imprimé

CMS

FPGA

Références



Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

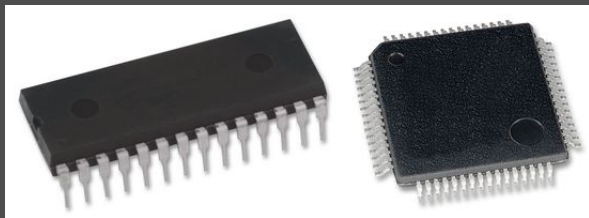
Arduino

Circuit imprimé

CMS

FPGA

Références



Le matériel

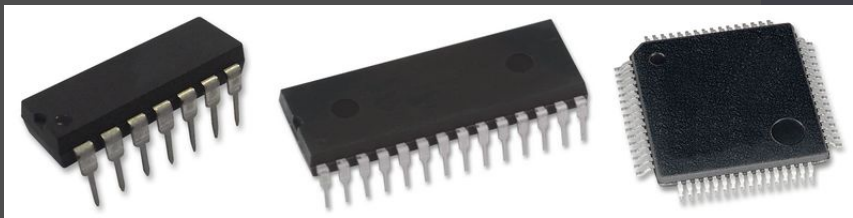
Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé



Le matériel

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références



Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

Arduino

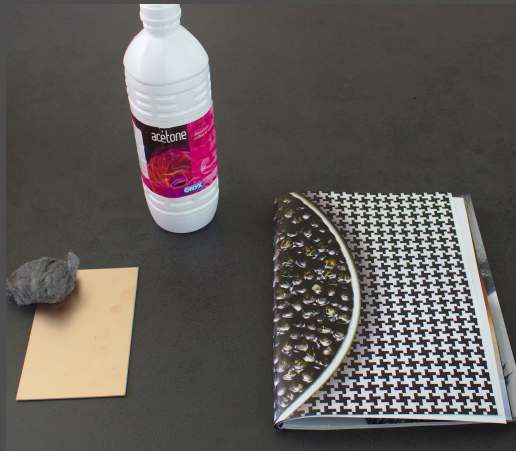
Circuit imprimé

CMS

FPGA

Références

Préparation



Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Préparation

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Transfert à chaud

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Transfert à chaud

Pushing (a little)
the limits
of your DIY
electronics



Sanpi

uction

no

t imprimé

nces

Transfert à chaud

Pushing (a little)
the limits
of your DIY
electronics

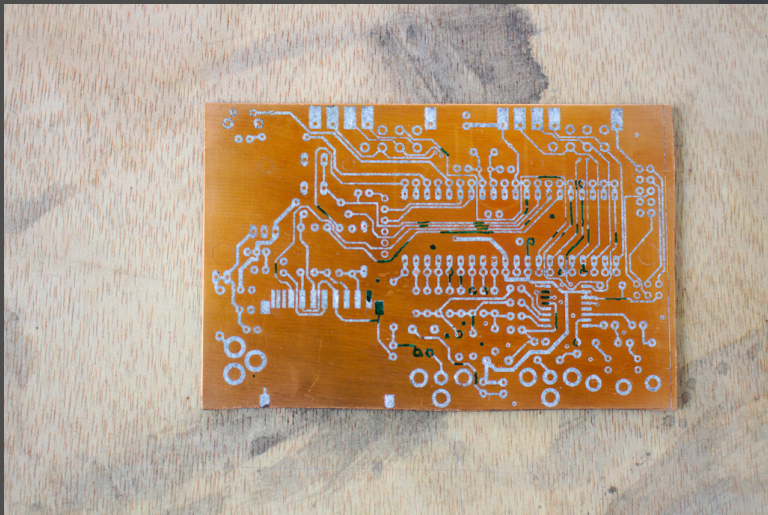
Sanpi

uction

to

t imprimé

nces



Transfert à froid

- ▶ Alcool 8/11
- ▶ Acétone 3/11

Gravure

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Gravure



Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Gravure

Pushing (a little)
the limits
of your DIY
electronics

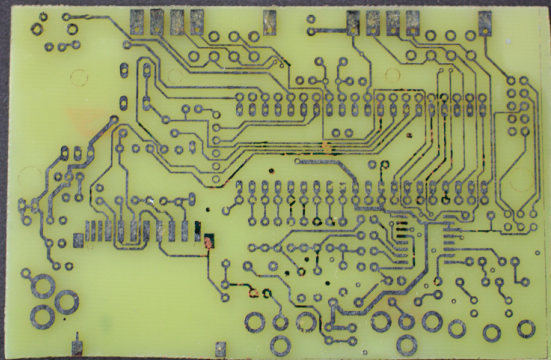
Sanpi

uction

no

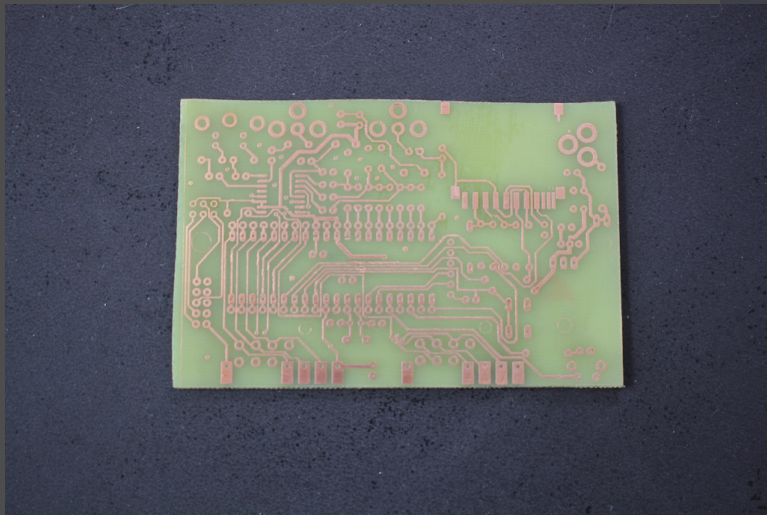
t imprimé

nces



Gravure

Pushing (a little)
the limits
of your DIY
electronics



Sanpi

uction

no

t imprimé

nces

Gravure

Pushing (a little)
the limits
of your DIY
electronics

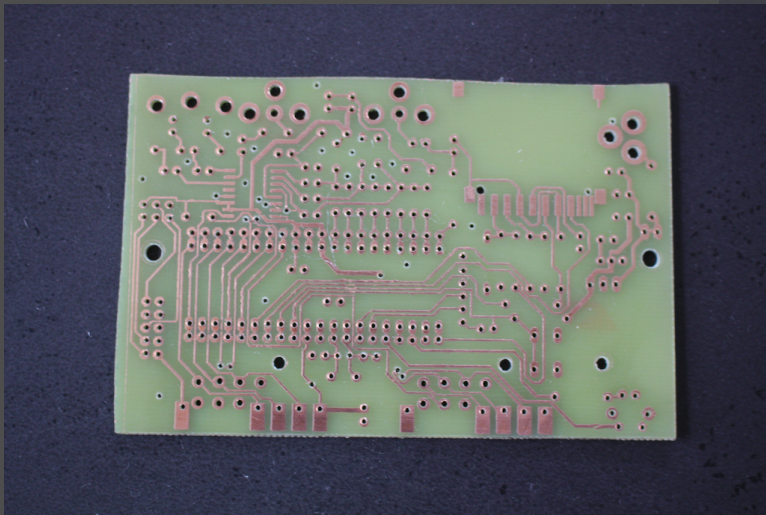
Sanpi

uction

no

t imprimé

nces



Gravure

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Pushing (a little)
the limits
of your DIY
electronics

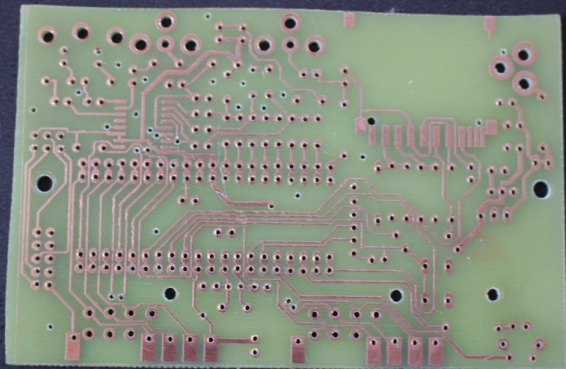
Sanpi

uction

no

imprimé

nces





Pushing (a little)
the limits
of your DIY
electronics

Sanpi

[Introduction](#)

[Arduino](#)

[Circuit imprimé](#)

[CMS](#)

[FPGA](#)

[Références](#)



Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références



Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

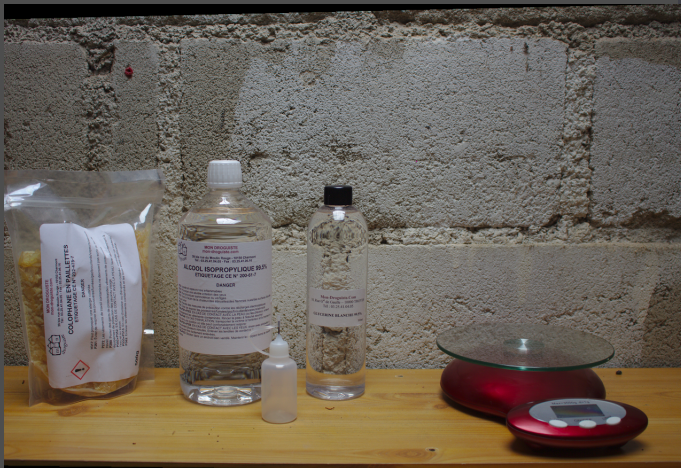
Arduino

Circuit imprimé

CMS

FPGA

Références





Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références



Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

FPGA

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références



FPGA

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

```
module blink(input clk, output led);  
  reg [31:0] cnt;  
  always @(posedge clk) cnt <= cnt+1;  
  assign led = cnt[22];  
endmodule
```

The screenshot displays the Vivado IDE interface for a project named "project_2". The main window shows the source file "blink.v" with the following Verilog code:

```

1 module LEDBlink(input clk, output LED);
2
3   reg [30:0] cnt;
4   always @(posedge clk) cnt <= cnt+1;
5
6   assign LED = cnt[22];
7   endmodule
8

```

The "Design Runs" table at the bottom shows the following data:

Design Run	Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed
synth_1	synth_1	constrs_1	Not started													
impl_1	impl_1	constrs_1	Not started													

The "Design Runs" table also includes columns for "WNS", "TNS", "WHS", "THS", "TPWS", "Failed Routes", "LUT", "FF", "BRAM", "URAM", "DSP", "Start", and "Elapsed". The "Start" and "Elapsed" columns are currently empty for the listed runs.

FPGA

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Outils libres

- ▶ yosys
- ▶ arachne-pnr
- ▶ icestorm

Icestudio

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Icestudio

Pushing (a little)
the limits
of your DIY
electronics

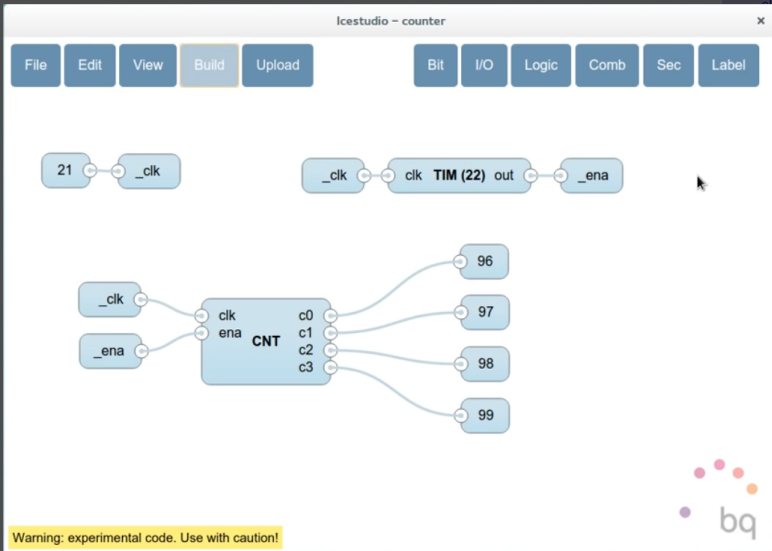
Sanpi

ction

a

imprimé

ces



Icestudio

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références

Sanpi

Introduction

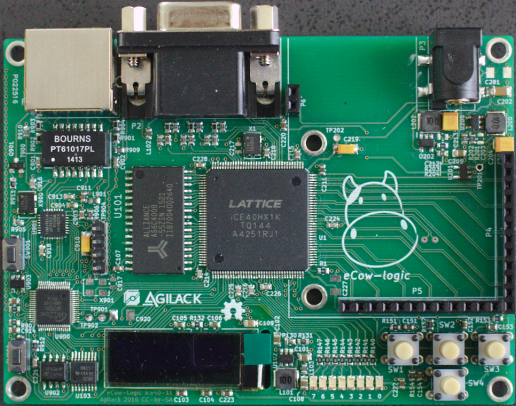
Arduino

Circuit imprimé

CMS

FPGA

Références



Sanpi

Introduction

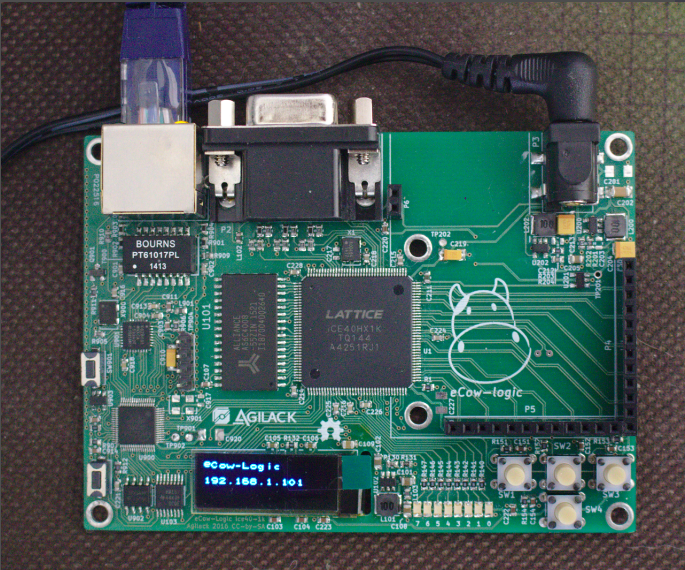
Arduino

Circuit imprimé

CMS

FPGA

Références



eCow-logic

Charger un bitstream

Fichier No file selected.

Envoyer



eCow-logic

Navigation



Fichiers >

k2000.v

k2000

ecow.pcf

k2000.v

Tools

Synthèse

Place / Route

Bitstream

Charge FPGA

Fermer l'éditeur

```
1 //
2 // eCow-logic - FPGA demo/example "K2000"
3 //
4 // Copyright (c) 2016 Saint-Genest Gwenael <gwen@agilack.fr>
5 //
6 // This file may be distributed and/or modified under the terms of the
7 // GNU General Public License version 2 as published by the Free Software
8 // Foundation. (See COPYING.GPL for details.)
9 //
10 // This file is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE
11 // WARRANTY OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
12 //
13 module top(
14     input  clk_i,
15     output led_0_o,
16     output led_1_o,
17     output led_2_o,
18     output led_3_o,
19     output led_4_o,
20     output led_5_o,
21     output led_6_o,
22     output led_7_o
23 );
24
25 wire [7:0]leds;
26
27 // Divide main clock to produce led clock
28 reg [24:0] counter;
29
30 always @(posedge clk_i)
31 begin
32     counter[24:0] <= counter[24:0] + 25'd1;
33 end
34
35 //
36 // State Machine
```

- ▶ `http POST 192.168.1.114/pld @blink.bin`

- ▶ http POST 192.168.1.114/pld @blink.bin
- ▶ démo ?

- ▶ https://media.ccc.de/v/camp2015-6634-pushing_the_limits_of_diy_electronics
- ▶ <http://oomlout.com/BBAC/src/guide/BBAC-01-guide-00ML-WEB.pdf>
- ▶ <http://www.nongnu.org/avr-libc/>
- ▶ <https://learn.sparkfun.com/tutorials/installing-an-arduino-bootloader>
- ▶ <https://www.arduino.cc/en/Hacking/Bootloader?from=Main.Bootloader>
- ▶ http://dangerousprototypes.com/docs/Bus_Pirate_AVR_Programming
- ▶ http://www.jtxp.org/tech/tinysafeboot_en.htm

Références

- ▶ https://fr.wikipedia.org/wiki/Syst%C3%A8me_g%C3%A9n%C3%A9ral_harmonis%C3%A9_de_classification_et_d%27%C3%A9tiquetage_des_produits_chimiques
- ▶ <http://wiki.electrolab.fr/Projets:Lab:2015:SolderStation:Manuel>
- ▶ <http://dangerousprototypes.com/blog/2012/06/14/workshop-video-how-to-make-a-simple-soldering-flux/>
- ▶ <http://www.instructables.com/id/Heatless-cold-Toner-Transfer-for-PCB-Making/>
- ▶ https://media.ccc.de/v/eh16-40-verilog_synthesis_and_more_with_yosys
- ▶ <http://www.clifford.at/yosys/>
- ▶ <http://www.clifford.at/icestorm/>
- ▶ <https://github.com/bqlabs/icestudio>
- ▶ <http://ecowlogic.fr/>

Questions ?

Pushing (a little)
the limits
of your DIY
electronics

Sanpi

Introduction

Arduino

Circuit imprimé

CMS

FPGA

Références



<https://github.com/sanpii/slides/releases/download/pses2016/push-limit-electronic.pdf>