

Incorporación de bootloaders a los PWRC2 / INTI-NQN 2021 para Laboratorio Cutral-Co

L&R Ingeniería - Rev. 2 08-03-2021 R. Oliva

1. RESUMEN

1.1 Incorporación de Bootloader a placas CL2 de los equipos PWRC2. Se continuó el trabajo iniciado en 2020, utilizando herramientas abiertas, para incorporar a las placas un mecanismo de bootloader por puerto serie a 115200 baud. Para ello se realizó la adaptación de dicha placa (que es contemporánea pero no nativamente "Arduino compatible") para su utilización con el Arduino IDE [ref0] y la función "quemar bootloader", utilizando una versión modificada de Optiboot de MightyCore [ref1] y el programa Avrdude [ref2]. Si bien la documentación y procedimiento debe depurarse, esto permite como se muestra en la Figura 1 acceder a las placas CL2, grabar el programa Optiboot en la zona especialmente reservada (accedida previamente al arranque del programa principal), con el clásico programador AVRISP Mkii con que cuenta el Laboratorio de INTI-Cutral Có.

PCF8563_EE_Test2 Arduince	1.8.12		AVR ISP
Archivo Editar Programa He	erramientas Ayuda		AVRISP mkll
CF8563_EE_Test2	Auto Formato Archivo de programa. Reparar codificación & Recargar. Administrar Bibliotecas Monitor Serie Serial Plotter	Ctrl+T Ctrl+Mayús+I Ctrl+Mayús+M Ctrl+Mayús+I	USBtinyISP ArduinoISP ArduinoISP.org USBasp Parallel Programmer Arduino as ISP
3 // R.Oliva 7.: 4 // Configurab 5 #include <rtc 6</rtc 	WiFi101 / WiFiNINA Firmware Updater ESP8266 Sketch Data Upload		Arduino as ISP (ATmega32U4) Arduino Gemma BusPirate as ISP
7 #include <eep 8 9 #define CL2_B 10 // Luego se de 11 Rtc_Pcf8563 R 12 13 bool hasPcf85 14 15 bool hasRtc =</eep 	Placa: "ATmega1284" Clock: "CL2bOsc 14.7456 MHz" BOD: "BOD 2.7V" Compiler LTO: "LTO disabled" Variant: "1284P" Pinout: "Standard pinout" Bootloader: "Yes(CL2 UART0)" Puerto Obtén información de la placa	Atmel STK500 development board Atmel JTAGICE3 (ISP mode) Atmel JTAGICE3 (JTAG mode) Atmel-ICE (AVR) Atmel ICE ISP Mode Atmel EDBG (Industruino SAMD) Atmel-ICE (Industruino SAMD) Atmel SAM-ICE (Industruino SAMD) STK500 as ISP (MightyCore) AVR ISP (MightyCore)	
16 bool rtcValid Porfavor, seleccione un pl	Programador: "AVRISP mkll (MightyCo Quemar Bootloader	re)" >	AVRISP mkll (MightyCore) USBtinyISP (MightyCore) ArduinoISP (MightyCore)
			ArduinoISP.org (MightyCore) USBasp slow (MightyCore) Parallel Programmer (MightyCore) Arduino as ISP (MightyCore) BusPirate as ISP (MightyCore) Atmel-ICE (AVR) (MightyCore) Atmel JTAGICE3 (ISP mode) (MightyCore) Atmel JTAGICE3 (ITAG mode) (MightyCore)

Figura 1 – Acceso previo al PWRC2 para grabación del bootloader Optiboot [ref1], utilizando ArduinoIDE

Como se observa, se utiliza una adaptación de las placas genéricas MightyCore para que CL2b aparezca en el menú, con la configuración de reloj, fusibles y demás parámetros específicos de dicha placa. Esta adaptación se irá mejorando para que resulte más directo el proceso de grabación del bootloader en todas las unidades PWRC2. Es importante hacer backup de Flash y EEPROM ya que la operación borra la memoria completa del controlador ATMega1284P [ref3]. Asimismo, se debe seleccionar AVRISP mkii (MightyCore), y recordar extraer la SD del equipo (como en los updates de Firmware) porque interfiere con la programación via el puerto SPI.

eófilo de Loqui 58 (fdo) 400 Rio Gallegos	L&R INGENIERIA	TE: 54 (0) 2966 430923 FAX: 54 (0) 2966 430923
Santa Cruz RGENTINA	www.lyr-ing.com	e-mail: roliva@lyr-ing.com

You created this PDF from an application that is not licensed to print to novaPDF printer (http://www.novapdf.com)



Hoja 2 de 7

1.2 Utilización del Bootloader en placas CL2 / PWRC2. Una vez finalizada la incorporación Optiboot, se apaga el PWRC2 y se retira el AVRISP, ya que la programación de Firmware ahora puede hacerse directamente vía puerto serie y a mayor velocidad (en forma similar a una placa Arduino Uno o Mega). Ahora solamente queda en memoria Flash el bootloader, no aparecerá display si encendemos nuevamente el PWRC2 el equipo queda esperando la carga de nuevo firmware con el protocolo STK500, y el OLED parpadea con 5 OLED blinks / seg, como se muestra en Figura 2. Aquí el ensayo es sobre un equipo similar (SISMED/SJ24) a los PWRC2 del INTI, y se supone que está conectado un conversor Serie/USB al COM principal del PWRC2 y conectado a la PC.



Figura 2 reencendido con bootloader Optiboot cargado, a la espera de nuevo Firmware vía puerto serie

Se copia el firmware original (aquí main1284.hex) realizado en C por el compilador Codevision AVR [ref1b], para el equipo SISMED/SJ24 (Figura 3), y se copia además el archivo send2.bat, que cumple funciones similares a lo que realiza internamente el Arduino IDE (llamando al programa Avrdude con una determinada configuración) para programar las placas basadas en procesador AVR. Además, se invoca la línea de comandos (cmd.exe) tanto en Win7 como en Win10, en el directorio en que se está trabajando, y se escribe "send2 main1284.hex", todavía sin presionar Enter:

* ^	Nombre	Fecha de modificación	Tipo	Tamaño	
*	main1284	22/4/2015 15:29	Atmel EEPROM co	1 KB	
*	main1284	22/4/2015 15:29	Intel HEX file	333 KB	
*	main1284	22/4/2015 15:29	Atmel FLASH cont	770 KB	
*	less send2	19/2/2021 09:02	Archivo por lotes	1 KB	
E COMPANY A LA CARA					
rosoft Windows\Systems rosoft Windows [2019 Microsoft	i2\cmd.exe Versión 10.0.18363.1379] Corporation. Todos los derecł	nos reservados.		-	>
CC\Windows\Systems crosoft Windows [) 2019 Microsoft \WorkCL2\2021\tes	22cmd.exe Versión 10.0.18363.1379] Corporation. Todos los derech ts\SISMED_SJ24_070321\Exe>ser	nos reservados. nd2 main1284.hex <u>.</u>		_	×
C:\Windows\System: crosoft Windows [) 2019 Microsoft .WorkCL2\2021\tes	<pre>i2\cmd.exe Versión 10.0.18363.1379] Corporation. Todos los derech ts\SISMED_SJ24_070321\Exe>ser igura 3 Firmware main1284.hex y a</pre>	nos reservados. nd2 main1284.hex_ archivo send2.bat copiados en el mi	ismo directorio, y llam	– ado a send2.	×

Teófilo de Loqui 58 (fdo) 9400 Rio Gallegos Santa Cruz ARGENTINA

www.lyr-ing.com

TE: 54 (0) 2966 430923 FAX: 54 (0) 2966 430923 e-mail: roliva@lyr-ing.com



Hoja 3 de 7

A continuación se presiona Enter, y después (pero casi en simultáneo) se enciende el PWRC2, con lo cual se debería comunicar el programa Avrdude que llama send2.bat con el Optiboot cargado en la placa CL2, e iniciar una secuencia de lectura/escritura/verificación como se muestra en la Figura 4.



Figura 4 Ejecución correcta de send2.bat para grabar el firmware nuevo en el PWRC2

A continuación se muestra un extracto de texto de la ventana de comandos, y la secuencia que ejecuta el avrdude para realizar la programación del equipo PWRC2:

C:\WorkCL2\2021\tests\SISMED_SJ24_070321\Exe>send2 main1284.hex

C: \WorkCL2\2021\tests\SISMED_SJ24_070321\Exe>C: \Users\Rafael \AppData\Local \Ardui no15\packages\ardui no\tool s\avrdude\6. 3.0-ardui no8/bi n/avrdude -CC: \Users\Rafael \AppData\Local \Ardui no15\packages\MightyCore\hardware\avr\2.0.3/avrdude.conf -v -patmega1284p -carduino -PCOM8 -b115200 -D -Uflash: w: main1284. hex: i

avrdude:	Version 6.3, compiled on Dec 16 2016 at 13:33:19
	Copyright (c) 2000-2005 Brian Dean, http://www.bdmicro.com/
	Copyright (c) 2007-2014 Joerg Wunsch

System	wide	configuration	file	is
"C:\Users\Rafael\AppData\	Local \Ardui no15\packages\	MightyCore\hardware\avr\2.0.3/avr	rdude.conf"	

I	Using Port			:	COM	8								
I	Using Programn	ner		:	ard	ui no								
	Overriding Bau	ıd Rat	te	:	115	200								
	AVR Part			:	ATm	ega12	84P							
(Chip Erase del	ay		:	550	00 us								
	PAGEL			:	PD7									
I	BS2			:	PAO									
I	RESET disposit	i on		:	ded	icate	d							
I	RETRY pulse			:	SCK									
:	serial program	n mode	÷	:	yes									
1	parallel progr	am mo	ode	:	yes									
-	Timeout			:	200									
:	StabDel ay			:	100									
	CmdexeDelay			:	25									
:	SyncLoops			:	32									
	ByteDelay			:	0									
I	PollIndex			:	3									
I	Pol I Val ue			:	0x5	3								
I	Memory Detail			:	:									
				Block	Poll			Page				P	olled	
	Memory Type	Mode	Delay	Size	Indx	Page	d Size	Si ze	#Pages	Mi nW	MaxW	Rea	dBack	
	eeprom			128			4096		0	9000	9000	Oxff	Oxff	
	flash	65	10	256	Ő	ves	131072	256	512	4500	4500	0xff	0xff	
	Lock	0	0	0	0	no	1	0	0	9000	9000	0x00	0x00	
	Ifuse	Ō	Ő	0	0	no	1	Ō	0	9000	9000	0x00	0x00	
	hfuse	0	0	0	0	no	1	0	0	9000	9000	0x00	0x00	
	efuse	Ō	Ő	0	0	no	1	Ō	0	9000	9000	0x00	0x00	
	signature	0	0	0	0	no	3	0	0	0	0	0x00	0x00	
	calibration	Ō	Ō	Ō	Ō	no	1	Ő	Ő	Ő	Ō	0x00	0x00	
ilo de Loqu	i 58 (fdo)				La	&R ING	SENIERIA				TE	: 54 (0)	2966 430) 23
) Rio Galle	jos										F/	AX: 54 (0) 2966 430)92

⁹⁴⁰⁰ Santa Cruz ARGENTINA

Teó

www.lyr-ing.com



Programmer Type : Arduino Description : Arduino Hardware Version: 3 Firmware Version: 8.0 Vtarget : 0.3 V Varef : 0.3 V Oscillator : 28.800 kHz SCK period : 3.3 us

avrdude: AVR device initialized and ready to accept instructions

avrdude: Device signature = 0x1e9705 (probably m1284p) avrdude: safemode: hfuse reads as 0 avrdude: safemode: efuse reads as 0 avrdude: reading input file "main1284.hex" avrdude: writing flash (121212 bytes):

avrdude: 121212 bytes of flash written avrdude: verifying flash memory against main1284.hex: avrdude: load data flash data from input file main1284.hex: avrdude: input file main1284.hex contains 121212 bytes avrdude: reading on-chip flash data:

avrdude: verifying ... avrdude: 121212 bytes of flash verified

avrdude: safemode: hfuse reads as 0 avrdude: safemode: efuse reads as 0 avrdude: safemode: Fuses OK (E:00, H:00, L:00)

avrdude done. Thank you.

Posteriormente, se apaga el PWRC2 y se vuelve a insertar la SD, y se abre el clásico Hyperterminal o Teraterm a 19200 baud y la secuencia es la habitual al encender el PWRC2 (ver listado y Figura 5):

```
Probando v1284P-SISM2_SJ24 15-03-2015
LCD Access testing..
-TestCtr=1 -TestCtr=2 -TestCtr=3 -TestCtr=4 -TestCtr=5 -TestCtr=6
Ahora RTC test..
RTC inicializando..
07/03/2021-21:16:05
Set CLKOUT to 1Hz
```

Ahora PSoC-M4E3 prueba de acceso (Read)..

M4EReadStatus_in = 0 M4EReadStatus_after = 0 bVect = 0 $bPW_0K_En = 1$ bBuzz = 0 bAux2 = 0 bLED = 0 bK_Aux = 0 IN0-P27 = 1 IN1-P25 = 0 IN2-P23 = 0 1N2:1N0 = 1 = 0 IN3-P21 IN4-P13 = 1 IN4: IN3 = 2 Teófilo de Loqui 58 (fdo)

9400 Rio Gallegos Santa Cruz ARGENTINA L&R INGENIERIA

www.lyr-ing.com

TE: 54 (0) 2966 430923 FAX: 54 (0) 2966 430923 e-mail: roliva@lyr-ing.com Hoja 4 de 7



T. DE LOQUI 58 - 9400 RIO GALLEGOS

Hoja 5 de 7

IN14-P20 = 1 IN15-P22 = 1 IN16-P24 = 0

..... (partes eliminadas, requiere reconfiguración)...

PreLg Archivo de Eventos: EA070321.CSV PostLg Archivo de Eventos: EA070321.CSV PW2 GIStatus find + Set Measurements to Start.. PW2 a) FRP Status: 3 b) IAE_DIPSw: c) VBat_DIPSw: 2 d) GlobalStatus: 3 Archivo de Eventos: EA070321.CSV SISM2 - Inicializacion Curva Potencia (Presione ? para comandos): ? - Listar comandos. 1 - Menu de Setup Prueba - Menu Setup Extendido 2 3 - Lanzar Prueba c/param. actuales (nuevo archivo) - Re-Lanzar Prueba (archivo en curso) 4 5 Configurar Canales



Figura 5 Vista de ejecución una vez programado del main1284.hex en un equipo SISMED similar al PWRC2, 7/3/21

1.3 Ventajas del Bootloader en placas CL2 / PWRC2. Se observa que el procedimiento facilita la actualización del firmware, que hasta el momento requería la remoción de la tapa transparente del PWRC2, la conexión del programador AVRISP/USBASP y la utilización de AVRStudio4 o similar para realizar los upgrades de Firmware, que habitualmente tardaba varios minutos. A través del Optiboot, y una vez que se concluya el proceso de mejora de parámetros de configuración e interfaz gráfica, el proceso como se ve del extracto se concluye en menos de 30 segundos.

Por otro lado, la utilización de la combinación CL2 y Optiboot se ha ensayado para la plataforma Arduino IDE más tradicional (que utiliza el lenguaje de programación simplificado (como se ve en el fondo de Figura 1). Sobre esto se está construyendo una nueva versión de PWRC2 abierta / OpenDLogger, de menor complejidad (el firmware actual del PWRC2 es de un tamaño medio pero supera las 41000 líneas de código C), que utilice programas clásicos (ejemplo test.ino), con sencillas secuencias setup() y loop(), además de muchas de las bibliotecas accesibles libremente de Arduino e implemente algunas propias que faciliten el acceso a los periféricos estándar de los PWRC2. Esto facilitará los ensayos muy específicos pero de corta duración y que puedan ser programados directamente por los ingenieros del INTI y modificados con cierta agilidad en comparación con la versión anterior.

La reciente adquisición en 2020 de Atmel (desarrollador de la línea de microcontroladores AVR) por parte de Microchip y la clara línea de sostenimiento de los modelos utilizados ATMega [ref3], que viene mostrando la empresa hace que resulte relevante

Teófilo de Loqui 58 (fdo)
9400 Rio Gallegos
Santa Cruz
ARGENTINA

L&R INGENIERIA

www.lyr-ing.com

TE: 54 (0) 2966 430923 FAX: 54 (0) 2966 430923 e-mail: roliva@lyr-ing.com



Hoja 6 de 7

sostener la actualización de este tipo de equipos, sobre todo en las actuales circunstancias económicas locales, por su costo reducido y disponibilidad de repuestos.

Asimismo se ha ensayado el uso de Optiboot/CL2 con la herramienta abierta PlaformIO [ref4], que facilita la programación en C++ en forma directa para muchos microcontroladores (además de la línea AVR, por ejemplo ESP8266 o ESP32) y se configura como plug-in en editores más avanzados como Atom o Visual Studio Code (VSC [ref5], gratuito). En la Figura 6 se aprecia el ensayo bajo Linux / Ubuntu 18.04 de una placa CL2 con la biblioteca LiquidCrystal de Arduino, utilizando VSC con el plug-in PlatformIO, y un display alfanumérico reducido de 16 x 2.

<pre> V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decided over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over Go Road Terminal Help V The first decide over first decide over first decide first first decide over first decide o</pre>	4			main.cpp - CL2Test1_LCDSerial4 - Visual Studio Code	
Image: 1 Properties Image: 2 Properies Image: 2 Propertie	0	File E	dit Selection View Go Run Terr	minal Help	
<pre> Product Tubes Pr</pre>	_			€ main.cpp x 🕸 platformio.ini 🛛 🥸 PIO Home	
<pre></pre>			✓ PROJECT TASKS	src > @- main.cop > 32 looo()	
<pre> Constant in the interval in the interval in the interval interval</pre>			> 🔄 Default		
 include a classificystal. As include a classif			✓ all env:ATmega1284P		
<pre> With the reduction plan number is consistent to updated best to der (LED Plan to P0, 5 on (L2) With the reduction plan number is consistent to 1 Work is biological plants With the reduction plan number is consistent to 1 With the reduction plan number is consistent to 1 Work is the reduction plan number i</pre>	0		✓ General	3 #include <liquidcrystal.h></liquidcrystal.h>	
<pre>> // The Light Section Light Hope Light Section Light Hope Light Section Light Se</pre>	-		O Build		
 i wooter <lii li="" wooter<=""> i wooter i wooter<</lii>			O Upload	5 // ATT LL2 Sketch with updated Bootloader (LEU Pin to PU.S on LL2) 5 // And MightyGare (ED pipout a manped to Cl2)	
Image: Second Secon			O Monitor		
<pre></pre>	-		 Upload and Monitor 	8 // initialize the library by associating any needed LCD interface pin	
 ✓ Putorm ✓ Putorm ✓ Program Save ✓ Original Arduno ✓ // Const in Fact, and Mathylor Std ✓ // Const in Fact, and Mathylor Std ✓ // Const in Fact, and Mathylor Std ✓ Machaded ✓ // Const in Fact, and Mathylor Std ✓ Machaded ✓ // Const in Fact, and Mathylor Std ✓ Machaded ✓ // Const in Fact, and Mathylor Std ✓ // Const in Fact, and Arduno ✓ Vol defact ✓ Vol	-0-		O Clean		
0 0 Program Size 11 // Const int rs = 13, en = 11, dn = 5, dn = 4, dn = 3, dn = 3, dn = 2; 0 Update Expension 13 // On CL 2 served 2.13 using HightyCore 51d 13 // On CL 2 served 2.13 using HightyCore 51d 13 // On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 13 // On CL 2 served 2.13 using HightyCore 51d 15 // On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 16 // On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 17 // On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 18 // On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 19 // On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 19 / On CL 2 served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 19 / Onder 2 // Or State Served 2.13 using HightyCore 51d 14 // On CL 2 served 2.13 using HightyCore 51d 10 // Dore 1 // State Hor P and B.13 using	A		✓ Platform		
Image: Construction 12 // 00 LC1 #3%-C.230 Buildy HightyCore 500 Image: Construction 13 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 14 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 15 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 16 // 00 LC1 #3%-C.230 Buildy HightyCore 510 Image: Construction 16 // 00 LC2 #3%-C.230 Buildy HightyCore 510 Image: Construction 16 // 00 LC2 #3%-C.230 Buildy HightyCore 510 Image: Construction 15 // 00 LC2 #3%-C.230 Buildy HightyCore 510 Image: Construction 16 // 00 LC2 #3% Construction 16 Image: Construction 16 // 00 LC0 Buildy Highty	_		Program Size		
0 9 Set Dates 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Num Botoldar 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Num Botoldar 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Num Botoldar 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Num Botoldar 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Num Botoldar 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Open Carbon 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Open Carbon 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Open Carbon 1 / 0 C1 20 Set Carbon 2 / 0 Set Set Duc Code here, to run once: 1 Open Carbon 1 / 0 C1 20 Set Carbon 1 / 0 C1 20 Set Carbon 0 Open Carbon 1 / 0 C1 20 Set Carbon 1 / 0 Set Set Duc Code here, to run once: 1 Set Duc Carbon 1 / 0 Set Set Duc Code here, to run once: 1 / 0 Set Set Duc Carbon 1 Set Duc Carbon 1 / 0 Set Set Duc Code here, to run once: 1 / 0 Set Set Duc Carbon 1 Deckes Collegin (12 A / 0 Set Set Duc Code Carbon 1 / 0 Set Set Duc C		0	O Upload EEPROM	12 // 0n CL2 rs=PC.2=18 using HightyCore Std	
0 Burn Backtader 15 // 0n C1 d derC 4-20 Using MightyCore Sid 0 Word Backtader 15 // 0n C1 d derC 4-20 Using MightyCore Sid 1 // 0n C1 d derC (-22 Using MightyCore Sid 16 // 0n C1 d derC (-22 Using MightyCore Sid 1 // 0n C1 d derC (-22 Using MightyCore Sid 16 // 0n C1 d derC (-22 Using MightyCore Sid 1 // 0n C1 d derC (-22 Using MightyCore Sid 16 // 0n C1 derC (-22 Using MightyCore Sid 1 // 0n C1 derC (-22 Using MightyCore Sid 16 // 0n C1 derC (-22 Using MightyCore Sid 1 // 0n C1 derC (-22 Using MightyCore Sid 16 22, d7 = 23, rw = 19; 1 // 0n C1 derC (-22 Using MightyCore Sid 16 22, d7 = 23, rw = 19; 1 // 0n C1 derC (-22 Using MightyCore Sid 16 20, dr = 23, rw = 19; 1 // 0n C1 derC (-22 Using MightyCore Sid 16 20, dr = 23, rw = 19; 1 // 0n C1 derC (-20 Using MightyCore Sid 16 20, dr = 23, rw = 19; 1 // 0n C1 derC (-20 Using MightyCore Sid 16 20, dr = 23, rw = 19; 1 // 0n C1 derC (-20 Using MightyCore Sid 16 20, dr = 23, rw = 19; 1 // 0n C1 derC (-20 Using Mi		-	O Set Fuses	15 // on CL2 energy, all using MightyCore Stdset to 8	
 > Advanced > Advanced > Advanced > Remote Development 			Burn Bootloader	15 // On CL2 d4=PC.4=20 using MightyGore Std	
<pre>> Remote Development</pre>			> Advanced		
iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii	9		> Remote Development		
Solution Solut				18 // On CL2 d7=PC.7=23 using MightyCore Std	
20 Clobel unit is = 1.5, un = 1.5, un = 1.5, un = 1.5, un = 1.5; un =				19 // So we re-write: 20 const. is the $c_{1} = 15 \text{ d} = 20 \text{ d} = 21 \text{ d} = 23 \text{ d} = 23 \text{ c} = 10$	
² POLKARCSS ² Void setup (1) { ² POLKONE				20 Constructed 1 - 10, en - 13, et - 20, ed - 22, ed - 22, et - 23, $(w - 2)$, $(w - 1)$,	
Solution 23 void setup() { Open 23 void setup() { Open 23 void setup() { Photocount 25 Serial.legin(13200); inspect 27 digitalWrite(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 29 lickoein(rx, UDW); // sets the digital pin as output inspect 20 lickoein(rx, UDW); // sets the digital pin as output inspect 20 lickoein(rx, UDW); // sets the digital pin as output inspect 20 lickoein(rx, UDW); // sets the digital pin as output inspect 20 lickoein(rx, UDW); // sets the digital pin as output inspect 20 lickoein(rx, UDW); // sets the din digital pin as output inspect	-		V QUICK ACCESS		
Open PD Account 24 PD Account // pit your setup code here, to run once: PD Account 25 PD Account 25 Protects K Configuration 26 Protects K Configuration Ubaries 29 Protects K Configuration 28 Protects K Configuration 28 Protects K Configuration V being 28 Protects K Configuration 28 Protects K Configuration 28 Protects K Configuration V being 29 Protects K Configuration 28 Protects K Configuration 10 Protects K Configuration V being Advanced Meerry Usage Is available via P Halforman elf Advanced Meerry Usage Is available via P Halformal Is available via P Halformal Is available via P Halformal Usage Is available via P Halformal Is available via P Halformal Usage Is available via P Halformal Is available via P Halformal Usage Is available via P Halformal Is available via	00			23 void setup() {	
PO AC const 25 Serial.Eggin(15200); Waset 27 djatalirite(rv, UUPU); // sets the digital pin as output Pipecta & Configuration 27 djatalirite(rv, UUPU); // sets the digital pin as output Devices Projecta & Configuration 29 1// sets the digital pin as output Pipecta & Configuration 29 1// sets the digital pin as output 1// sets the digital pin as output Devices Projecta & Configuration 29 1// sets the digital pin as output 1// sets the digital pin as output Devices Projecta & Configuration 29 1// sets the digital pin as output 1// sets the digital pin as output Devices Projecta & Configuration 29 1// sets the digital pin as output 1// sets the digital pin as output Devices Orcelains Source A flatforms Projecta & Configuration 1// sets the digital pin as output Sate Debugo Drosole Projecta & Configuration 1 - 66 (used 26 bytes from 13044 bytes) 1// source 20 bytes 1// source 20 bytes Platform Douber V updates 1 - 66 (used 26 bytes from 13044 bytes) 1 - 66 (used 23 bytes from 13044 bytes) 1 succeeded in 60:00:06:06.23	-		Open		
Notest 20 pinwose (Mr, 00000); // sets the duplate opin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Configuration 20 idjuitivite(mr, Low pin of f Notest & Status Notest & Status Notest & Status Notest & Concol Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & Status Notest & S			PIO Account	25 Serial.begin(19200);	
Projects & Configuration 20 27 site up the CONFiguration 20 107 site up the CONFiguration 100 site up the CONFiguration 100 site up the CONFiguration 1000			Inspect	26 pinMode(W, OUIPUI); // sets the digital pin as output 27 digitalWrite(w, UN); // sets the pulpin off	
Ubravies 29 Lfd.bejin(16, 2):			Projects & Configuration	28 // set up the LCD's number of columns and rows:	
Badds 30 // Prifit a message to the LO. Platforms Devices Collection concol Teaminal lable via 'PlatformIO Home > Project Inspect' Advanced Memory Usage Is available via 'PlatformIO Home > Project Inspect' Advanced Memory Usage Is available via 'PlatformIO Home > Project Inspect' Start Debug Console Platform 1.16k (usad 26 bytes from 13004 bytes) Togle Debug Console Platform ID Home > Project Inspect' Whit: 1.10k (usad 25 bytes from 13004 bytes) Togle Debug Console Platform Usage ISAB Whit: 1.2.9 (used 3728 bytes from 13004 bytes) Togle Debug Console Platform Usage ISAB Whit: 1.2.9 (used 3728 bytes from 13004 bytes) Platform Updates Environment Status Duration Hittorm Update All Terminal will be reused by tasks, press any key to close it. Platform Core CU Usad Kall @0 4.0 K R & C K			Libraries	29 lcd.begin(16, 2);	
mainting Plations procuss curve cu			Boards		
Devices Checking size_pip/buildATmegal280/fit means elf ✓ Debug Satt Oebugging Togle Debug Console Feasting size_pip/buildATmegal280/fit means elf Ø Pistors Bibray Updates Environment Statt Oebugging Togle Debug Console Ø Pistors Ø Pistors Ø Modelleneous Pistors Opdate Bibray Updates Environment Statte Debug Concel Inspect* Ø Modelleneous Pistors Opdate Bibray Ø Ø Ø Ø Ø Modelleneous Pistors Opdate Bibray Ø Ø Ø Ø Ø Ø Ø Ø Ø Ø Ø Ø Ø	101000		Platforms	PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL	1: Task - Build (ATmega 🗸 🕂 🗍 🏦 🏠
 > > >			Devices	Charking size nig/build/ATmons1204D/firmusra alf	
Start bebagging v Update W Updates PAH: [] 1.6% (used 266 bytes) 'n 10384 bytes) ' 2.9% (used 266 bytes) ' 1.9% (used 266 bytes) ' 2.9% (used 266 bytes) ' 1.9% (used 266 bytes	1		✓ Debug	Advanced Memory Usage is available via "PlatformID Home > Project Inspect"	
Togele Debug Console PLass: [] 2,2% (used 3/28 bytes from 1.90448 bytes) [SUCCESS] Took 8.23 seconds With any Updates Platerm Updates Vector Platerm Updates Platerm Updates Vector Environment Status Duration With any Updates Platerm Updates Vector Annega1284P Success I succeeded in 60:00:60.23 With any Updates Platerm Updates Vector Terminal will be reused by tasks, press any key to close it. I succeeded in 60:00:60.23 With any Updates Vector Vector I succeeded in 60:00:60:23 OB Vector I succeeded in 60:00:60:23	-		Start Debugging	RAM: [] 1.6% (used 268 bytes from 16384 bytes)	
 v Updates v Updates v Environment Status Durary Updates Environment Status Durary Updates V Environment Status Durary Updates V Environment Status Durary Updates V Environment Status Durary Updates Status <			Toggle Debug Console	Flash: [] 2.9% (used 3/28 bytes from 130048 bytes) [SUCCESS] Took 8.23 seconds	
Bit op back All Environment Status Duration Pittorm Update All Armospillable OP dat All Armospillable OP dat All Terminal will be reused by tasks, press any key to close it. Puttorm Update All Instal Gal Space: 2 UT SelectEncoding multiple OP dat All Instal Gal Space: 2 UT SelectEncoding multiple Image: All Plate All Pla			 Updates 		
Platermic polates Update All Platermic Cor CLI Amega1284P Succession 00:00:00:00:233 **** /********************************			Library Updates	Environment Status Duration	
Update All 1 succeeded in 00:00:06.233 Image: Subset of the state of the sta			Platform Updates	ATmega1284P SUCCESS 00:00:08.233	
			Update All		
			 Miscellaneous 	Terminal will be reused by tasks, pross any key to close it	
			PlatformIO Core CLI		
#1 문 2015 일 램 보 등 1 ~ 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2		⊗0⊿	∆0 ଲ ✓ → 🛢 🕈 🗵		Ln 43, Col 2 Spaces: 2 UTF Select Encoding rmIO
패 수 밝 🕐 🔚 💼 🌀 😵 🥍					🔛 🕑 🛄 🚭 🖉 🔚 🔛 🚱 💭 🖘
	-		# 💽 🧮 🛱 💼 🄇	🖗 🍄 🎽	스 惊고 아이 ESP 18:01 2020001

Figura 6 Utilización de Visual Studio Code y el plug-in PlatformIO bajo Linux con una placa CL2 y Optiboot conectada (02-21)

-0-

1.4 Referencias

- [ref0] https://www.arduino.cc/en/software
- [ref1] https://github.com/MCUdude/MightyCore
- [ref1b] https://www.codevision.be/
- [ref2] https://www.nongnu.org/avrdude/
- [ref3] https://www.microchip.com/wwwproducts/en/ATmega1284
- [ref4] <u>https://platformio.org/</u>
- [ref5] https://code.visualstudio.com/

Fecha de Revisión: Marzo 8 - 2021

Teófilo de Loqui 58 (fdo) 9400 Rio Gallegos Santa Cruz ARGENTINA L&R INGENIERIA

TE: 54 (0) 2966 430923 FAX: 54 (0) 2966 430923 e-mail: roliva@lyr-ing.com

You created this PDF from an application that is not licensed to print to novaPDF printer (http://www.novapdf.com)

www.lyr-ing.com

NOTAS:

L&R INGENIERIA www.lyr-ing.com TE: 54 (0) 2966 430923 FAX: 54 (0) 2966 430923 e-mail: roliva@lyr-ing.com

You created this PDF from an application that is not licensed to print to novaPDF printer (http://www.novapdf.com)