



مقدمه:

در جلسه قبل با وسایل ورودی خروجی استاندارد یک سیستم کامپیوتری کوچک از قبیل LCD کاراکتری ۲x۱۶ و کی پد ماتریسی ۴x۴ آشنا شدیم. در این جلسه با ابزارهای I/O پیشرفته تر از قبیل LCD گرافیکی تک رنگ 2-TS12864A برای نمایش تصاویر و فونت های فارسی و همچنین کیبرد کامپیوتر برای ورود اطلاعات حرفی / رقمی آشنا می شویم. در نهایت به معرفی سنسور دمای LM35 می پردازیم که برای ساخت دماسنج های ساده در محدوده دمای اتاق به کار می رود و با معرفی رله و نحوه راه اندازی بارهای ۲۲۰ ولت پروژه کنترل دمای هوشمند را طراحی و کد نویسی می کنیم.

اصولاً برنامه نویسی در محیط بسکام به دلیل دارا بودن کتابخانه های متنوع ساده بوده و کاربر نیازی به مطالعه دیتا شیت ها و پروتکل های ارتباطی وسایل جدید ندارد؛ البته به همین دلیل برای ارتباط با هر وسیله ای وجود کتابخانه آن نیز ضروری است و این امر کاربر را محدود به استفاده از ابزارهای خاص می کند (مثل LCD گرافیکی 128 در 64 مدل KS0108 که در ایران طرفداران بیشتری دارد). در زبان های اسمبلی و C هر چند کاربر نیاز به یادگیری پروتکل های ارتباطی تمام دستگاه ها دارد ولی محدودیتی نداشته و با توجه به رایگان بودن کامپایلرهای GCC و نیز اسمبلر و کامپایلر معروف AVR Studio به راحتی می توان پروژه هایی با قابلیت های منحصر به فرد ایجاد کرد. به آموزش اسمبلی و C نزدیک می شویم... درجلسه پنجم به مدیریت وقفه ها و کار با تایمرها و همچنین پورت سریال و شماره گیری تلفن و ... در محیط بسکام می پردازیم و از جلسه **ششم به بعد** کار با C و اسمبلی (زبان مادری میکرو) را آغاز می کنیم.

ارتباط با LCD گرافیکی مدل TS12864A-2 (چیپ KS0108)

نمایشگر مذکور که تصویر آن را در شکل مقابل مشاهده می کنید به صورت تک رنگ و در اندازه های مختلفی تولید می شود. مدلی که در اینجا بررسی می کنیم نوع ۱۲۸ در ۶۴ پیکسل با بکلایت آبی می باشد. این نمایشگر ۲۰ سیم ارتباطی دارد و تغذیه آن ۵ ولت مستقیم است. پایه های این نمایشگر به سه دسته تقسیم می شوند: ۱- پایه های تغذیه، کنتراست و بکلایت؛



۲- پایه های Data، ۳- پایه های کنترل. هشت پایه دیتا و شش پایه کنترل به صورت زیر به میکروی نمونه ATmega16/32 متصل می شوند؛ (پایه های دیتا را به ترتیب ۰ تا ۷ به هر پورتی می توان وصل کرد و پایه های کنترل به هر ترتیب دلخواه به هر پورت دلخواه حتی به پایه های پورت های مختلف قابل اتصال اند) تصویر زیر مربوط به یکی از پروژه های رایگان AVR64.com برای رسم نمودار است.



برنامه ميكرو

پس از اتصال LCD نوبت به تعریف نمایشگر برای میکرو و آشنایی با توابع آن می رسد . همانند پروژه های جلسات قبل یک فایل جدید در بسکام ایجاد کرده و کدهای زیر را در آن وارد کنید. از این به بعد می توانید با وارد کردن خطوط ۱ تا ۶ در متن برنامه از دست تنظیمات Option تقریباً راحت شوید. خطوط ۴ تا ۶ به یک مقاله جدا احتیاج دارند و جزء نکات خاص بسکام هستند. فعلاً مقادیر این ۳ خط را روی ۶۴ قرار دهید و برای میکروهای کوچک مثل Tiny13 که Ram کوچکی دارند مقدار ۱۶ برای هر یک کفایت می کند، برای میکرو های بزرگ مثل M128 و با توجه به تعداد فراخوانیها و سابروتین ها و متغیرهای محلی استفاده شده در برنامه بهتر است مقدار هر یک را روی ۱۲۸ تنظیم نمایید. این مطلب در جای خود تحت عنوان پشته ها توضیح داده خواهد شد.

🔚 BAS(:OM-AVR IDE [1.11.9.1] - [D:\BackUp\	oite\##AVR64##\Article\Basco	m ,Code ,Winavr ,Fast\AVR4	\GLCD.bas] 🔤 🗗 🔀					
🔣 Eile	<u>E</u> dit <u>V</u> iew <u>P</u> rogram <u>T</u> ools <u>O</u> ptions	<u>W</u> indow <u>H</u> elp		11 12 1_8 ×.					
	📙 🖳 🍐 🙆 😫 🛯 🐰 🐚 📋 💷	年 🔎 🧏 . 🛛 🍐 🗹 🐐 🌾	🔟 • 🖉 🛅 📮 🛲 💷 🖄	- 100 ? - 26.					
GLCD.b	GLCD.bas 🖾 Paint.bas 9 3 💌								
Sub		▼ Label							
	1			Package DIP40					
	4 Shwstack = 64 5 Sswstack = 64 5 Sframesize = 64								
1	\$ \$lib "glcdKS108 lbx" Config Graphlcd = 128 * 64s PORTC , Ce = 4 , Ce2 = 3 ,	ed , Dataport = PORTB , Cd = 0 , Rd = 1 , Rese	Controlport = et = 5 , Enable = 2						
1	2 Setfont Font8x8								
1	4 Cls 5 Lcdat 1 , 1 , "Salam"								
1	/ 8 End 9 \$include "font8x8.font"								
222	1 2 3			зтрял					
2222	* 5 6 7			Package : DIP40 , Pin : 24					
223	3 9 0			PC2 - Generic IO pin PC2 TCK - JTAG - Test Clock.					
<			>	synchronous to TCK.					
34: 1	Modified Insert								
🦺 st	art 🔜 🔛 BASCOM-AVR 📄 2 Wir	dows 👻 🜆 2 Microsoft	Windows Med EN	Sign 🦥 🛞 🎯 🙆 4:11 PM					

توجه: LCD به کار رفته در این پروژه فقط با مفهوم Pixel آشناست و برای نوشتن عبارات و کاراکترها نیاز به فایل فونت مورد نظر داریم، فایل فونت font8x8.font را می توانید از آدرس زیر واقع در پوشه مثال های بسکام کپی گرفته و در پوشه برنامه جاری ذخیره نمایید؛ (مثالهای بسکام در ورژه جدید در my document ذخیره می شود)





پس از مونتاژ مدار روی برد بورد و پروگرم کردن میکرو خروجی مقابل را خواهید داشت، اگر چیزی روی LCD مشاهده نکردید پتانسیومتر کنتراست را اندکی بچرخانید. اگر با خطای File not مواجه شدید معنایش اینست که فایل فونت را کنار فایل برنامه کپی نکرده اید. این برنامه نسبتاً ساده است و تنها دستور کاربردی آن دستور

Lcdat 1,1,"Salam"

است. در این دستور دو عدد اول موقعیت سطر و ستون مکان نما را مشخص می کنند و در قسمت سوم هم ر شته حرفی، متغیر و یا لیست ترکیبی از متغیر ها و حروف که با ; از هم جدا شده اند نوشته می شود.



اگر در آخر دستور بالا یک عدد ۱ اضافه کنیم (Lcdat 1,1, "Salam", 1) خروجی روی نمایشگر به صورت انتخاب شده در می آید و برای ساختن منوهای انتخابی کاربرد زیادی دارد که با ترکیب آن با صفحه لمسی خروجی جالبتری به دست می آید.



نکته ای که در اینجا باید به آن اشاره کنیم اینست که تا اینجا همه چیز مثل LCD کاراکتری معمولی ۲ در ۱۶ بود، ولی تفاوت مهم این نوع LCD عدم شناسایی فونت هاست، به همین جهت مجبوریم که فایل فونت ساخته شده توسط MCS Elec را در کنار برنامه کپی کنیم. این کار یک مزیت نسبت به معمولی ۱۶ در ۲ دارد و آن توانایی تغییر نرم افزاری فونت هاست. همچنین می توانیم فایل فونت فارسی بسازیم و با کپی گرفتن آن در کنار برنامه در نمایشگر فارسی بنویسیم. در اینجا فایل فونت infont8x8.font را با ++Notepad باز می کنیم تا ببینیم داخل آن چیست.



در داخل فایل فونت ۸ در ۸ مشاهده می کنیم که چیزی جز یک جدول دیتا وجود ندارد. در این جدول که با زبان اسمبلی نوشته شده است در ابتدا نام داخلی فایل آمده که ما با دستور Setfont Font8x8 در ابتدای برنامه آن را انتخاب کرده ایم. سپس علامت شروع کدهای اسمبلی و به دنباله آن کدهای مزبور به وضعیت روشن و خاموش بودن پیکسل های هر کاراکتر. با توجه به ۸ در ۸ بودن مربع

های فرضیِ در بر گیرنده هر کاراکتر، برای هر خط، از ۸ عدد احتمالاً ۸ بیتی استفاده شده است. برای مطمئن شدن فرضیات توجه خود را روی کاراکتر علامت تعجب ^۱؛ متمرکز می کنیم. دیتاهای آن را مجدداً در خط پایین می نویسیم:

.db 0,0,6,95,6,0,0,0 ; !

برای بررسی دقیق تر، کاراکتر '!' را روی LCD نمایش می دهیم و وضعیت پیکسل های آن را با عبارت بالا بررسی می کنیم:



همانطوریکه در تصاویر بالا مشاهده می فرمایید اگر کاراکتر'! را داخل یک مربع ۸ در ۸ که از گوشه بالا و سمت چپ نمایشگر شروع شده در نظر بگیریم اعداد داخل خط بالا به ترتیب از چپ با این مربع تطبیق دارند. به ازای دو عدد ۰ که باینری آن ها 00000000 می شود دو ستون ۸ تایی خاموش داریم. عدد بعدی ۶ است که باینری آن می شود: 00000110 و دقیقاً دو پیکسل روشن و ۶ پیکسل خاموش در ستون سوم از سمت چپ داریم. عدد بعدی ۹۵ است که باینری آن 111111 میشود و می بینیم که ۶ پیکسل روشن و ۲ پیکسل خاموش داریم. بس اعداد موجود در هر خط وضعیت روشن و خاموش بودن پیکسل های نمایشگر را نشان می دهند. به همین ترتیب با دستکاری اعداد موجود در فایل فونت می توانیم شکل و شمایل کاراکترها را تغییر دهیم. البته این کار وقتگیری است و MCS راهکار بهتری را ارائه داده است که در ادامه با آن آشـنا مـی شویم.

طراحی فونت با Plug In کاربردی Font Editor

Font Editor یک پلاگ این جدا برای طراحی و ویرایش فونت می باشد. برای نصب این برنامه از منوی Tools روی Tools Manager کلیک کرده و چک باکس Fonteditor.bpl را تیک بزنید. در Plugin Manager که چنین چیزی وجود نداشت بایستی آن را از یکی نسخه های بسکام کپی کرده و در پوشه Plugins واقع در محل نصب برنامه ذخیره نمایید. پس از تیک زدن این گزینه و زدن دکمه OK یک بار بسکام را ببندید و مجدداً باز کنید، سپس از منوی Tools روی Plugins و سپس روی Font Editor کلیک نمایید. پنجره و مجدداً باز کنید، سپس از منوی Font Editor مطابق شکل زیر باز می شود:

B	₽ • ७) =	A	WR_4.docx - Microso	oft Word			x
Ho	💐 MCS Font E	ditor	- Mailings Da	Nizere			0
	File				9		
V Clipboard	Fontname	Width	Height	From 32	to 8	1 = Change Stilling Styles = 4	
		_				7 5	」 [5]
1							
•		_				طرا	
1.1						·	
2 - 1 -						برناه	
-						نیک	
						و در	
4						itor	
2				_		C	
9	Preview	Clear all Clipbrd			+		
		Set all Suggest		+	+		
- 					+		*
							⊙ ∓
4						•	
Page: 6 of 7	Words: 1,321	🥸 Persian			1 🗔 🚍 🔳 93%		Ð
🏄 start	2 B.	🕇 🛅 3 W 🔸 🕼 2 M. 🔸	📀 Wi 🚯 2 B	. 👻 📝 No	🦉 2 P. 👻 EN	Sign 🎽 🕜 👂 5:17 PM	

در این برنامه می توانیم یک فایل فونت جدید با Size های مختلف و زبانها و کاراکترهای متفاوت بسازیم، البته کار کردن با فونتهای فارسی فوق العاده مشکل است چرا که به طور مثال حرف "ف" سه کاراکتر مختلف "ف"، "ف" و "ف" دارد که برای هرکدام باید یک کاراکتر طراحی کنیم و با توجه به توابعی که حروف قبل و بعد از حرف مورد نظر را بررسی می کنند تصمیم گیری کنیم که کدامیک را نشان دهیم. دقیقاً همان کاری که Editor موبایل موقع تایپ SMS با سرعت انجام می دهد! (این برنامه برای پروژه تابلو روان فارسی با MMC در 3000 خط(!) نوشته شده و در بخش پروژه ها برای دانلود قرار داده شده است).

برای ساخت یک فونت جدید از منوی فایل روی New کلیک کنید و فیلدهای پنجره باز شده را

New Fo	nt 💶 🗖 🔀	ئمے OK
Font name	AVR64	خـاب هـر
Height	8 🏂	.شــاهده و
Width	8	لام" را بـر
Max ASCII	255	های بـلا
	Color	کتر هـای
		، صورت با
L	V Ok X Cancel	ارت

مطابق شکل روبرو تنظیم نمایید. پس از زدن دکمه OK پنجره اصلی Font Editor نمایان شده و با انتخاب هر کاراکتر و زدن کلید Suggest می توانید آن را مشاهده و ویراش کنید. ما در اینجا تصمیم داریم عبارت "سلام" را بر روی نمایشگر بنویسیم. برای این منظور از کاراکترهای بلا استفاده ۱۳۰ به بعد بهره می گیریم و آنها را با کاراکتر های "س"، "ل"، "لا" و "م" جایگزین می کنیم. در این صورت با نوش

Chr(133);Chr(132);Chr(131);chr(130) رشته "سلام" را بر روی نمایشگر خواهیم داشت.

برای شروع مطابق شکل زیر روی کاراکتر ۱۳۰ کلید کرده و دکمـه Suggest را فشـار مـی دهـیم، سپس به طراحی کاراکتر "سـ" می پردازیم.

AVR_4.docx - Microsoft Word						⊂ x					
9	Home	Insert	Page Lavout	References	Mailings	Review	View			1 12 1	0
1	X Ar	📣 MCS Fo	nt Editor [AV	R64.font]						An 2	44
		File							9		3
Paste	🛷 📴	Fontname	AVR64	Widtł	8	Height 8	Fr	om 32	to 255 8	Change E Styles *	diting
Clipbo	ard 🐨	88011	Char 🔼						7		
E 1.1	18 • • • 17 • 2	130								1111112	· · 🖏
-		111									
. 6 .		132									
-		133									
1		134									
-11-		135									
12 - 1		136									
-		131									
- 13		138									
14											
-		Descience	Clear								
- 1		Freview									
16 -			Set al	I Suggest				-	→		
1.2									,		1
-											۲
1	-										Ŧ
Page: 7	7 of 8 Wo	ords: 1,542	🥸 Persian) 🛱 🛃 🗇	93% 🕞)——————————————————————————————————————	-+
🏄 s	tart	2 B	🗣 🛅 3 W	• 💓 2 Mi	Vine	d 👔 2	2 Bl	🦉 2 P	- EN Sign		19 PM

پس از اتمام کار فایل را با نام AVR64 در کنار فایل برنامه ذخیره می کنیم و برنامه را مطابق زیر تغییر می دهیم. (ما در اینجا از نام یکسان برای نام داخلی فایل و نام خارجی فایل استفاده کردیم که در این صورت دستورات Setfont و Sinclude نام یکسانی را Add می کنند، توجه داشته باشید که در جلوی setfont نباید پسوند فایل فونت را بنویسیم).

🔚 BASCOM-AVR IDE [1.11.9.1] - [D:\BackUp\Site\##AVR64##\Article\Bascom,Code,Winavr,Fast\AV	/R4\GLCD.bas] 📃 🖃 🔀
Eile Edit View Program Tools Options Window Help	11 ¹² /1 - ₽ × -
] 🔌 🖟 📲 🍈 🖉 🖬 🗸 🐚 🖺 標 標 🖉 🦂 . 🧄 🗹 🦠 📲 🛯 🖉 🛲 💷	
GLCD.bas 🖾	9 Y 3 🖂
Sub Label	SChip PinOut 및 × 공
<pre> 1 Sregfile = "m32def.dat" 2 Scrystal = 100000 3 4 Shwstack = 64 5 Sswstack = 64 6 Sframesize = 64 7 8 Slib "glcdKS108.lbx" 9 Config Graphled = 128 * 64sed , Dataport = PORTB , Controlport = _ 10 PORTC , Ce = 4 , Ce2 = 3 , Cd = 0 , Rd = 1 , Reset = 5 , Enable = 1 12 Setfont Avr64 13 14 Cls 15 Lcdat 1 , 1 , Chr(133) ; Chr(132) ; Chr(131) ; Chr(130); 16 17 18 End 19 Sinclude "AVR64.font" 21 22 23 24 25 26 27 28 29 30 30 30 </pre>	2 Rackage DIP40 Search Search Package : DIP40, Pin : 24 PC2 - Generic IO pin PC2 TCK - JTAG - Test Clock, JTAG operation is synchronous to TCK.
32: 2 Modified Insert	
- 🏄 start 🛶 🔚 BASC 🗧 🗀 3 W 🖌 🞯 2 Mi 🗸 📀 Wind 😵 2 Bl 👻 🦉 2 P 🔸	EN Sign 🔌 🔇 🐉 5:52 PM

اگر همه چیز به خوبی پیش رفته باشد خروجی زیر را خواهید داشت:



بدین ترتیب با استفاده از ۱۲۷ گزینه خالی پایین کد های ASCII می توانید کاراکترهای دلخواه خود را طراحی کرده و با دستورات (Chr(index کاراکتر مورد نظر خود را نمایش دهید. البته ممکن است ایجاد کاراکترهای بیشتر از این هم ممکن باشد که در صورت نیاز باید آزمایش کنید ولی به هر حال از ۸ کاراکتر LCD Designer نمایشگرهای ۱۶۲۲ بیشتر است. (با C قضیه راحت تر می شود چون ممکن است استاندارد اسکی در برخی موارد خاص به ۱۲۸ کاراکتر محدود شود و ۲۵۶ کاراکتری نباشد.)

دستورات گرافیکی

تا اینجا فقط از قدرت نمایش متن LCD گرافیکی استفاده کردیم. ولی مهمترین و جذاب ترین ویژگی این نوع LCD ها قابلیت نمایش عکس های تک رنگ با رزولیشـن ۱۲۸ در ۶۴ پیکسل و نیـز قابلیت ترسیم خطوط و دایره و... می باشد که همگی از ویژگی آدرس پذیر بودن تـک تـک Pixel هـا حاصـل مـی شـود. در آموزش های C و WinAVR خواهیم دید که چگونه موقع نوشتن هدر فایـل هـای C از کـوچکترین ویژگـی شروع کرده و با انجام و محاسبات ریاضی و توابع تو در تو امکانات جالبی را می آفرینند. به طور مثال از ویژگی آدرس پذیر بودن یک پیکسل می توان تابعی نوشت که شعاع و مرکز دایره را گرفته و با محسبات ریاضی ماننـد یک پرگار به سرعت پیکسل می توان تابعی نوشت که شعاع و مرکز دایره را گرفته و با محسبات ریاضی ماننـد درمورد رسم خط بین دو نقطه که احتمالاً در دوران راهنمایی و دبیرستان با جدول مختصات دکارتی آشنا شده اید. البته به این نکته توجه داشته باشید که مختصات اکثر دستگاه های الکترونیکی از قبیـل Touch و محـور درمورد رسم خط بین دو نقطه که احتمالاً در دوران راهنمایی و دبیرستان با جدول مختصات دکارتی آشنا شده عمودی محور لاهایی دکارتی نیست و ۲ آن در گوشه بالا و سمت چپ آن است. محـور افقـی محـور X هـا و محـور



از ویژگیهای جالب LCD گرافیکی قدرت انتخاب یک پیکسل و روشن و خاموش کردن آن است که با دستور Pset x, y, 0 و Pset x, y, 255 انجام می شود. دستور اول پیکسل موجود در آدرس x, y را خاموش می کند و دستور دوم آن را روشن می نماید. از این دستور در ساخت پروژه اسیلوسکوپ که برای دانلود در بخش پروژه های سایت AVR64.com قرار داده شده است بهره فراوان برده ایم.

دستور بعدی دستور 255 , (x2 , y2) - (Line(x1 , y1) است. این دستور خطی بین دو نقطه داده شده رسم می کند، مانند دستور قبل اگر عدد ۲۵۵ را به ۰ تغییر دهیم خط مورد نظر پاک می شود. از این دستور نیز برای ساخت پروژه رسم نمودار روی LCD گرافیکی استفاده شده است. دستورات دیگری نیز مانند Circle و… وجود دارند که کار با آنها ساده بوده و روزبه روز بر تعداد آنها افزوده می شود. برای مشاهده جزئیات بیشتر به Help بسکام مراجعه کنید تا از موضوع اصلی منحرف نشویم و قدری در مورد نمایش عکس روی نمایشگر صحبت کنیم. توجه داشته باشید که با قابلیت نمایش عکس می توانید متون زیادی در Paint ویندوز به صورت فارسی نوشته و بدون نگرانی از عدم Support فونت بر روی نمایشگر نشان دهید.

نمایش عکس

یکی از قابلیت های منحصر به فرد نمایشگرهای گرافیکی امکان نمایش عکس های تک رنگ می باشد. در این نوع LCD امکان نمایش عکس هایی به اندازه ۱۲۸ در ۶۴ پیکسل به صورت سیاه و سفید وجود دارد. برای ساخت یک عکس با ایان ویژگی ها وارد Paint ویندوز شده و از منوی Image روی گزینه Attributes کلیک کنید. در پنجره باز شده تنظیمات را مطابق شکل زیر انجام دهید و کلید OK را بزنید.

Attributes	? 🔀
File last saved: Not Available Size on disk: Not Available Resolution: 64 x 64 dots per inch Width: 128 Height: 64 Units O Inches O Cm O Pixels	OK Cancel Default
Colors Image: State of the state of th	

در پنجره جدید Paint می توانید تصاویر و نوشته های فارسی و انگلیسی و نیز منوها و آیتم هایی را که قصد نمایش آنها را در پروژه مورد نظر دارید طراحی کرده و در کنار فایل برنامه ذخیره نمایید.



به محیط BASCOM باز می گردیم و برای تبدیل عکس ذخیره شده به فرمت فشرده از منوی Tools روی Graphic Converter کلیک می کنیم. در پنجره باز شده در بخش LCD Type گزینه 128*64 را انتخاب کرده و در سمت راست <u>گزینه SED Series را تیک می زنیم</u>، سپس با فشردن دکمه Load عکس مورد نظر را که با پسوند bmp ذخیره کرده بودیم باز کرده و با زدن دکمه Save آن را به نام Picture.bgf در پوشه برنامه ذخیره می نماییم. در نهایت روی دکمه OK کلیک می نماییم.

BASCOM-AVR IDE [1.11.9.1] - [D:\BackUp\Site\##AVR64##\Article\Bascom,Code,Winavr,Fast\AVR	4\GLCD.bas] 📃 🗗 🗙						
🔀 File Edit View Program Tools Options Window Help	1 1 1_ # X.						
<u></u>							
GLCD.bas 🖾 Paint.bas	9 💙 3 👳						
Sub	sChip PinOut 및 × 및						
1 \$regfile = "m32def.dat" 2 \$crystal = 1000000 3	Package DIP40						
A Shwstack = 64 Srwstack = 64 Sframesize = Stib "glcdKS10 Ocofig Graphic Deorric , Ce = If Deorric , Ce = If Deorric , Ce = If Setfont Avr64 Avr64 Avr64 Sinclude "Avr64 Deorric and a structure Save Deorric and a structure Deorric and	Package : DIP40, Pin : 24 PC2 - Generic IO pin PC2 TCK - JTAG - Test Clock. JTAG operation is synchronous to TCK.						
32: 2 Modified Insert							
🔧 start 💦 BASCOM 🖻 3 Wind 🗸 🐏 AVR_4.d 😵 IVT Corp 🦉 Picture.b	EN Sign 🎽 🎯 🚯 10:07 PM -						

هنگامی که عکس مورد نظر را باز می کنیم از فرمت BMP به فرمت BGF که یک فایل باینری خاص و فشرده است تبدیل می شود. فایل جدید حجم کمی داشته و با کامپایل کردن برنامه در قسمتی از حافظه فلش به صورت جدول دیتا ذخیره می شود. استفاده زیاد از عکس موجب پر شدن سریع حافظه فل شده و فضای برنامه را هدر می دهد، سعی کنید به جای عکس ها از دستورات خط و پیکسل برای ساختن اشکال هندسی استفاده کنید؛ کاری که در پروژه پایانی این مقاله انجام داده ایم.

پس از تبدیل و ذخیره عکس، برنامه را به صورت زیر تغییر می دهیم. در برنامه جدید فایل فونت ها را از حالت Include خارج کرده و به جای آن تصویر مورد نظر را Add می کنیم.

🔀 BASCOM-AVR IDE [1.11.9.1] - [D:\BackUp\Site\##AVR64##\Article\Bascom,Code,Winavr,Fast\A	VR4\GLCD. bas] 📃 🖃 🔀
Ele Edit View Program Iools Options Window Help	
GLCD.bas 🖾 Paint.bas	9 3 🗹
Sub	8 ^{Chip PinOut} 및 × 및
<pre> State 1 Sregfile = "m32def.dat" 2 Scrystal = 1000000 3 Shvstack = 64 5 Ssvstack = 64 6 Sframesize = 64 7 8 Slib "glcdKS108.lbx" 9 Config GraphIcd = 128 * 64sed . Dataport = PORTB . Controlport = 10 PORTC . Ce = 4 . Ce2 = 3 . Cd = 0 . Rd = 1 . Reset = 5 . Enable = 11 12 Cls 13 Shovpic 0 . 0 . Pic 14 15 End 16 Pic: 17 Shgf "Picture.bgf" 18 19 20 21 22 23 24 24 25 26 27 28 29 30 </pre>	2 Package DIP40 Search Search Search Package DIP40 Package : DIP40, Pin : 24 P22 - Generic IO pin PC2 TCK - JTAG - Test Clock, JTAG operation is synchronous to TCK.
35:1 Modified Insert	
🛃 Start 🚽 🔚 BASCOM-AVR 📄 🎦 Windows E 👻 🗐 AVR_4.docx 💲 IVT Corporatio	EN Sign 🎽 🎯 🚯 10:17 PM

با پروگرم کردن آی سی خروجی زیر را خواهیم داشت. در این برنامه چون سایز عکس متناسب با طول و عرض LCD انتخاب شده فقط مجاز به نمایش آن در مختصات ۰ و ۰ هستیم. در صورتی که عکس های کوچک ۸ در ۸ طراحی کنید می توانید آنها را در هر مختصاتی قرار دهید. (X بین ۰ تا ۱۲۷ ولی Y به صورت ۸



تایی یعنی ۰، ۸، ۱۶، ۲۴، ۳۲، ۴۰، ۴۸ یا ۵۶ چرا که مختصات ۷ موقع نمایش عکس و نیز هنگام نمایش متن، بیت آدرس پذیر نیست و بایت به بایت عوض می شود که به نوع آدرس دهی LCD مربوط می شود و تنها دستور Pset است که هم x و هم ۷ آن بیت به بیت و از ۰ تا ۱۲۷ و ۰ تا ۶۳ آدرس پذیرند. البته دستور Line و Circle هم همینطورند و برای

اطلاعات دقیقتر بهتر است به Help بسکام رجوع کنید و نیز با آزمایش و خطا دریابید.

ورود اطلاعات (کیبرد کامپیوتر)

تا اینجا با نحوه نمایش اطلاعات بر روی خروجی آشنا شدیم، LCD گرافیکی مانند مانیتور یک سیستم کامپیوتری کوچک قادر به نمایش اطلاعات مختلفی است. اما در طرف دیگر برای ورود اطلاعات نیاز به ابزاری داریم که کمی از کی پد ۴ در ۴ پیشرفته تر باشد و تفاوت آن را بتوان مانند تفاوت بین LCD معمولی ۲X۱۶ و GLCD درک کرد. برای ورود راحت تر اطلاعات در یک سیستم کامپیوتری بهترین وسیله کیبرد کامپیوتر است. این ابزار اطلاعات را به صورت سریال و تنها با دو رشته سیم به میکرو تحویل می دهد. یک رشته کلاک و رشته سیم دیگر دیتا. خوشبختانه کتابخانه کیبرد کامپیوتر در Bascom موجود بوده و ما درگیر جنبه های فنی این وسیله نخواهیم شد.

برای ارتباط با کیبرد کامیپوتر بهتر است در ابتدا با پایه های آن آشنا شوید. در تصویر زیر مشاهده می کنید که پایه های ۱ مربوط به دیتا، ۳ زمین، ۴ تغذیه ۵ ولت و ۵ کلاک می باشند. برای راه اندازی کیبرد پایه های زمین و + آن را به تغذیه میکرو متصل می کنیم و پایه های دیتا و کلاک را نیز به دو پایه دلخواه از میکرو مثلا PD.0 و PD.1 وصل می کنیم.



4

(At the computer) 6 PIN MINI-DIN FEMALE (PS/2 STYLE) at the computer. Pin Name Dir Description DATA 🕶 Key Data 1 2 n/c Not connected 3 Gnd GND -4 Power . +5 VDC VCC -5 CLK -Clock 6 n/c Not connected

ما در اینجا پایه شماره ۵ یعنی Clock را به Pind.0 میکرو و پایه شماره ۱ یعنی Data را به Pind.1 میکرو و پایه شماره ۲ یعنی Pind.1 میکرو وصل کرده ایم. پایه های ۳ و ۴ را هم به زمین و ۵ ولت وصل می کنیم. توجه داشته باشید که

در برنامه نام پایه کلاک نوشته نمی شود و مستیماً پس از دستور Config می آید ولی نام پایه دیتا نوشته می شود.



بسکام را باز کنید و کد های زیر را در آن وارد نمایید. (نظر به اینکه تمام کـدها در عکـس زیـر معلـوم نیستند در صفحه بعد متن کامل کدها را آورده ایم).



پس از پروگرم کردن میکرو و اتصال کی برد، تغذیه را وصل می نماییم. ممکن است یک لحظـه تمـام



پان روز را را را ی ی روز را ی ی روز را ی ی ر LED های کیبرد چشمک بزنند که در کیبردهای مختلف این قضیه فرق می کند. پس از شروع تایپ خروجی زیر را خواهیم داشت. پس از اینکه صفحه پر شد یک لحظه پیغام !FULL نمایش داده می پر شد یک لحظه پیغام !Home نمایش می گردد. می سان داد می کاراکت با Eptor

توانید از عدد کاراکترهای Enter و Tab و Delete استفاده کنید و بسته به نوع پروژه کار های متف اوتی را مثل ورود اطلاعات، تایید نهایی، پاک کردن کاراکتر قبلی، رفتن به خط بعدی و انجام دهید. البته این برنام هنوز با یک ادیتور واقعی فاصله زیادی دارد.

```
متن کامل کدهای مربوط به راه اندازی کیبرد کامپیوتر و نمایش نوشته ها بر روی نمایشگر گرافیکی در
زیر آورده شده است:
```

```
$regfile = "m32def.dat"
$crystal = 1000000
Shwstack = 64
\mathbf{\hat{S}}swstack = 64
Sframesize = 64
$lib "glcdKS108.lbx"
Config Graphled = 128 * 64sed , Dataport = Portb , Controlport = _
Portc , Ce = 4 , Ce2 = 3 , Cd = 0 , Rd = 1 , Reset = 5 , Enable = 2
Setfont Font8x8
Config Keyboard = Pind.0 , Data = Pind.1 , Keydata = Keydata
Dim A As Byte , S As String * 1 , Row As Byte , Col As Byte
Row = 1 : Col = 1
Cls
Do
   A = Getatkbd()
    S = String(1, A)
    If A <> 0 Then
       Lcdat Row , Col , S
Col = Col + 8
       If Col > 128 Then
           Col = 1
           Incr Row
           If Row > 8 Then
              Cls
              Lcdat 4 , 50 , "FULL!"
              Waitms 500
              Cls
              Row = 1 : Col = 1
           End If
       End If
   End If
Loop
End
$include "font8x8.font"
Keydata:
 'normal keys lower case
Data 0, 44, 107, 105, 111, 48, 57, 0, 0, 46, 45, 108, 48, 112, Data 0, 0, 0, 0, 0, 92, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 92, 0, 0
Data 0, 60, 0, 0, 0, 92, 0, 0, 0, 0, 13, 0, 0, 92, 0, 0
Data 0, 60, 0, 0, 0, 0, 8, 0, 0, 49, 0, 52, 55, 0, 0, 0
Data 48, 44, 50, 53, 54, 56, 0, 0, 0, 43, 51, 45, 42, 57, 0, 0
                                                                       108 , 48 , 112 , 43 , 0
'shifted keys UPPER case
```

ضمناً نیازی به مقاومت های pull-up که در ویرایش قبلی گفته شده بود نیست. این مقاومت ها برای شبیه سازی کیبرد توسط AVR لازم است. (شبیه سازی کیبرد عکس قضیه بالا بوده و می توان با AVR کی برد DV طراحی کرد. مناسب برای کمپانی هایی که قصد دارند کیبرد کامپیوتر تولید کنند).

پروژه دماسنج دیجیتال هوشمند

در اینجا کار ما با سیستم های ورودی و خروجی اطلاعات به پایان می رسد و کار روی پروژه پایانی این بخش یعنی دماسنج دیجیتال هوشمند را آغاز می کنیم. البته از ابزار و آموخته های بالا (بجز کیبرد کامپیوتر) در این پروژه استفاده خواهیم نمود. در این پروژه با سنسور دمای LM35 و کنترل بار ۲۲۰ ولت توسط رله آشنا می شویم همچنین می آموزیم که چگونه از مبدل آنالوگ به دیجیتال داخلی میکرو استفاده نماییم.

سنسور دمای LM35 قطعه ای شبیه به ترانزیستور BC547 می باشد که ساده ترین و ارزانترین سنسور دما می باشد و در مدلهای مختلفی عرضه می شود. هر کدام از این مدلها یک پسوند مثل A، D یا... دارند که محدوده عملکرد آنها را مشخص می کند. برخی از این سنسورها دمای منفی را نیز اندازه گیری می کنند و رنج برخی از آنها بین ۵۵- تا ۱۵۰+ است ولی تمام آنها محدوده دمای نرمال اتاق یعنی ۰ تا ۳۰ را شامل می شوند که ما از همین محدوده در این پروژه استفاده خواهیم کرد و یک دماسنج دیجیتال ۰ تا ۱۰۰ درجه با قابلیت کنترل دو رله در دو دمای تنظیم شده حد پایین و حد بالا به همراه نمایشگر آنالوگ گرافیکی طراحی می کنیم.

0.01 نظر به اینکه LM35 یک سنسور آنالوگ بوده و به ازای هر درجه سانتیگراد ولتاژ خروجی آن 0.01 ولت افزایش می یابد بایستی از مبدل آنالوگ به دیجیتال داخلی میکرو برای تبدیل این مقدار آنالوگ به عدد استفاده نماییم. مبل آنالوگ به دیجیتال میکرو با دقت ۱۰ بیت کار می کند و در صورتی که ولتاژ مرجع آن روی ۵ ولت تنظیم شود به ازای ۵ ولت ورودی مقدار ماکسیمم یعنی ۱۰۲۳ را نمایش می دهد. ما به دمای بین روی ۵ ولت تنظیم شود به ازای ۵ ولت ورودی مقدار ماکسیمم یعنی ۱۰۲۳ را نمایش می دهد. ما به دمای بین م درجه ۱۰ درجه نیاز داریم و با توجه به اینکه این سنسور به ازای هر درجه 0.01 ولت خروجی می دهد پس در مع درجه درجه دولت و در ۱۰۰ درجه ۱ ولت خواهیم داشت. یعنی ما از محدوده ۰ تا ۵ ولت میکرو فقط به محدوده ۰ تا ۱ ولت آن نیاز داریم و اگر یک تناسب ببندیم میکرو در ۱ ولت عدد ۲۰۰ را به ما می دهد که با تقسیم آن بر ۲ عدد ۱۰۰ را خواهیم داشت و این عدد نهایی همان دمای محیط است. پس در این حالت، دماسنجی با دقت ۱ را درجه خواهیم داشت و این عدد نهایی همان دمای محیط است. پس در این حالت، دماسنجی با دقت ۱ درجه خواهیم داشت و این عدد نهایی هر درا ولت تنظیم کنیم داشت. درجه خواهیم داشت در درجه در ۱ ولت میکرو فقط به محدوده ۰ تا ۱ ولت آن نیاز داریم و اگر یک تناسب ببندیم میکرو در ۱ ولت عدد ۲۰۰ را به ما می دهد که با تقسیم آن بر ۲ عدد ۱۰۰ را خواهیم داشت و این عدد نهایی همان دمای محیط است. پس در این حالت، دماسنجی با دقت ۱ درجه خواهیم داشت و این عدد نهایی همان دمای محیط است. پس در این حالت، دماسنجی با دقت درجه را خواهیم داشت و این مد نهایی همان دمای محیط است. پس در این حالت، دماسنجی با دقت درجه را خواهیم داشت و این می کنیم.

5 V	1023	
1 V	?∼200	

در شکل زیر یک ترتیب پایه ها و ولتاژ مصرفی این سنسور را که ۵ ولت می باشد ملاحظه می فرمایید. پایه وسط این سنسور به طور مستقیم به یکی از پایه های مبدل A2D میکرو که روی پورت A هستند وصل می شود. ما در این پروژه سنسور را به مبدل شماره صفر یعنی پایه ۴۰ میکرو وصل می کنیم.



قبل از رسم شماتیک، به توضیح آخرین قطعه به کار رفته در این پروژه می پردازیم. برای کنترل بار ۲۲۰ ولت یا کمتر به یک قطعه الکترومکانیکی به نام رله احتیاج داریم. این قطعه چیزی جز یک سیم پیچ و یک کلید مغناطیسی بالای آن نیست. مزیت این سیستم عایق بودن بار خروجی نسبت به ورودی است. مطابق شکل زیر هرگاه سیم پیچ با ولتاژ کم (مثالاً 5 ولت) آهن ربا شود تیغه را جذب نموده و می تواند یک بار قوی تر و یا هم اندازه با خود ولی عایق نسبت با ورودی را قطع و



وصل نمايد.



در تصویر بالا یک رله را به همراه مدار داخلی آن ملاحظه می فرمایید. در این رله ۵ ولت پایه های A و B ورودی سلف هستند و پایه های C و D کانکتورهای خروجی. چنین رله ای برای راه اندازی معمولاً به جریانی در حدود ۱۰۰ میلی آمیر نیاز دارد که خارج از توان یک پایه میکروست؛ بنابراین برای Drive کردن یک رله از مداری مطابق شکل زیر استفاده می کنیم. ترانزیستور به کار رفته معمولاً BC547 و مقاومت نیز 4,7kΩ می باشد.



پس از توضیح قطعات پروژه دماسنج هوشمند نوبت به ترسیم شماتیک و برنامه نویسی آن می رسد. در شکل زیر شماتیک پروژه دماسنج دیجیتال هوشمند را ملاحظه می فرمایید:



همانطوریکه در شماتیک بالا ملاحظه می فرمایید دو رله ۵ ولت به همراه متعلقات به پایه های PINA.6 و PINA.7 متصل شده اند. یک عدد سنسور دمای LM35 نیز به پایه شماره ۴۰ میکرو یعنی PINA.0 که همان مبدل آنالوگ به دیجیتال داخلی شماره ۰ میکرو می باشد متصل شده است. دیود های IN4.14 که همان مبدل آنالوگ به دیجیتال در مقابل پسماند ناشی از قطع و وصل بارهای سلفی رله ها قرار 1N4148 برای محافظت از مدار دیجیتال در مقابل پسماند ناشی از قطع و وصل بارهای سلفی رله ها قرار داده شده اند. بخش سلف و خازن پایین شماتیک نیز برای تغذیه ADC در نظر گرفته شده و از ورود نویز به این بخش حساس میکرو تا حد قابل توجهی جلوگیری به عمل می آورد.

نوبت به برنامه نویسی پروژه دماسنج هوشمند می رسد:



با توجه به اینکه تمام کد در تصویر بالا مشخص نیست متن کامل کد را در صفحه بعد آورده ایم. اولین بخش جدید این کد دستور Donfig ADC است که مبدل ADC داخلی میکرو را پیکره بندی می کند، آخرین آرگومان این دستور ولتاژ رفرنس را مشخص می کند و برای ۵ ولت از AVCC استفاده می کنیم. دو دستور بعدی ADC را روشن و نمونه برداری را آغاز می کنند. در قسمت بعد LCD را پیکره بندی کرده و سپس پایه های رفته به رله ها را به صورت خروجی پیکره بندی می کنیم. این دو دستور اسمبلی هستند و فرقی با دستورات LCD این دستور اسمبلی هستند و فرقی با دستورات LCD و می کند. در بخش بعدی با دستورات خط و دایره دور نمایشگر BGF یک خط کشیده و عکس یک دماسنج را در وسط آن رسم می کنیم. این دستورات به مراتب کمتر از عکس BGF فضا می گیرند و قابلیت حمل برنامه را افزایش می دهند چرا که فایل BGF را نمی توان با فایل FUF جابجا کرد! دستور (0) می ایشگر نشان داده می شود. دستورات بعدی دستورات شرطی هستند که با توجه به دمای Min و Max رله ها را که به گرم کننده و سرد کننده می روند کنترل می کنند. ایـن سیسـتم می تواند دمای یک محیط صنعتی مثل گلخانه و یا یک خانه معمولی را بخوبی کنترل کند.

سورس كامل پروژه دماسنج دیجیتال هوشمند + تصویر خروجی:

```
$regfile = "m32def.dat"
$crystal = 1000000
hwstack = 64
\$swstack = 64
framesize = 64
Config Adc = Single , Prescaler = Auto , Reference = Avcc
Enable Adc
Start Adc
$lib "glcdKS108.lbx"
Config GraphIcd = 128 * 64sed , Dataport = Portb , Controlport = _
Portc , Ce = 4 , Ce2 = 3 , Cd = 0 , Rd = 1 , Reset = 5 , Enable = 2
Setfont Font8x8
Dim Temp As Word
sbi Ddra,6
sbi ddra,7
Cls
Cls
Circle(80, 40), 5, 255
Line(78, 34) -(78, 10), 255
Line(82, 34) -(82, 10), 255
Line(78, 10) -(82, 10), 255
Line(78, 10) -(82, 10), 255
Line(1, 1) -(126, 1), 255
Line(126, 1) -(126, 62), 255
Line(126, 62) -(1, 62), 255
Line(1, 62) -(1, 1), 255
Do
      Temp = Getadc(0)
     Temp = Temp / 2
Lcdat 2 , 8 , Temp ; "'C "
Lcdat 6 , 8 , "Min: 20"
Lcdat 7 , 8 , "Max: 35"
      If Temp > 35 Then Sbi Porta , 7 If Temp < 35 Then Cbi Porta , 7
      If Temp < 20 Then Sbi Porta , 6
      If Temp > 20 Then Cbi Porta , 6
      Waitms 100
Loop
End
$include "font8x8.font"
```



در این جلسه کار با دستگاه های ورودی و خروجی استاندارد مانند LCD گرافیکی تک رنگ و کی برد کامپیوتر را آموختیم و نیز از مبدل A2D میکرو جهت نمایش دما استفاده کردیم. همچنین نحوه کنترل بار ۲۲۰ ولت توسط رلـه آموزش داده شد. بسکام استاندارد قابلیت های زیادی دارد و در صورت تمایل می توانیـد نسخه کامـل آن را بـه قیمـت حدود ۱۳۴ هزار تومان از سایت تولید کننده آن به نشـانی mcselec.com توسط ویـزا کـارت خریـداری نماییـد. (آپدیت 90: نمایندگی رسمی بسکام در ایران: embed.ir و دفتر فروش yazdkit.com ایند در صورتی که می توانید پول آن را با انجام چند پروژه به دست آورید این کار را انجام دهید. قابلیت های منحصر به فرد بسکام عبارتنـد از دستورات PTP/IP و کار با شبکه و اینترنت و نیز دستورات SOG و کار با هارد کـامپیوتر و کـارت MIR و... البتـه این دستورات در قالب هدر فایل های رایگان C نیز در دسترس است و برای علاقه مندان C و کامپایلر WinAVR و... البتـه این امکانات به صورت رایگان فراهم شده است. اگر شما از آن دسته افرادی هستید که با نسخه های کامل ولی غیر مجـاز این امکانات به صورت رایگان فراهم شده است. اگر شما از آن دسته افرادی هستید که با نسخه های کامل ولی غیر مجـاز این امکانات به صورت رایگان فراهم شده است. اگر شما از آن دسته افرادی هستید که با نسخه های کامل ولی غیر مجـاز این امکانات به صورت رایگان فراهم شده است. اگر شما از آن دسته افرادی هستید که با نسخه های کامل ولی غیر مجـاز دردیداری نشده) بسکام کار می کنید طبق قوانین کپی رایت اگر کار تجاری با این برنامه انجام دهید و پول قابل تـوجهی از این محصول بدست آورید باید یک نسخه از آن را خریداری نمایید. امروزه این کار به راحتی توسط شـرکت هـای ارائـه دهنده خدمات مستر/ ویزا کارت امکان پذیر است. (هم اینک خرید آن از داخل ایران ممکن شده است).

آموزش ویژگیهای یک نرم افزار و در مقابل پیشنهاد استفاده نکردن از آن کمی ضد ونقیض به نظر می رسد، ولی همانطوریکه در ابتدای این سری مقالات گفته شد هدف اصلی نمایش امکانات میکروکنترلر و در نهایت آموزش زبان C و اسمبلی است. به همین روی در جلسه بعدی (جلسه پنجم) در مورد تایمرها و کانترها و ویژگیهای آنها از قبیل (خروجی PWM، شماره گیری تلفن [DTMF] ، ساعت داخلی RTC، پورت سریال RT2–23 یا همان USART) و نیز مدیریت و کاربرد وقف های میکرو صحبت می کنیم. بسکام فقط کار با امکانات میکرو را کمی ساده تر می کند ولی در عوض جزئیات داخلی را از چشم کاربر وقف های میکرو صحبت می کنیم. بسکام فقط کار با امکانات میکرو را کمی ساده تر می کند ولی در عوض جزئیات داخلی را از چشم کاربر دوقف های میکرو صحبت می کنیم. بسکام فقط کار با امکانات میکرو را کمی ساده تر می کند ولی در عوض جزئیات داخلی را از چشم کاربر دور می نماید. اگر برنامه نویسی خود را تقویت کنید کد نویسی در محیط نت پد به زبان اسمبلی یا محیط کد ویژن و WinAVR هیچ تفاوتی با Bascom و یا و یا و ASC یا PHP نخواهد داشت و فقط ساختارها هستند که کمی تغییر می کندد. – از جلسه شیم اسمبلی یا محیط کر و یا و ASC یا PHP نخواهد داشت و فقط ساختارها هستند که کمی تغییر می کند. – از جلسه شیم اسمبلی یا محیط کر و یا کمی تغییر می کند. – از جلسه شیم اسمبلی یا محیط که که کمی تغییر می کند. – از جلسه شیم اسمبلی یا محیط که کمی تغییر می کند. – از جلسه ششم اسمبلی را شروع می کنیم –

ادامه دارد ...

(برای تولید سورس های بسکام / بیسکام از نسخه دموی Bascom 1.11.9.1 استفاده شده است)

(انتشار این مقاله آزاد بوده و برای تولید آن از نسخه خریداری شده ویندوز XP و MS Word 2007 و نسخه رایگان CutePDF استفاده شده است، AVR64 به شدت تابع قوانین کپی رایت بین المللی بوده و تحت هیچ شرایطی از نرم افزار های غیر قانونی استفاده نمی کند)

> مولف: بهنام زکی زاده <u>www.avr64.com</u> ۳۱ مرداد ۱۳۸۹

آخرین ویرایش ۱۳ مهر ۱۳۹۲