【カメラを作るぞ】プロジェクト

2023年4月26日

事前チェック

- 1. ino ファイルを作成する。
- 2. スケッチ例(ESP32->camera->cameraWebServer)の

cameraWebServer.ino app_httpd.cpp camera index.h

camera pins.h

をコピーする。

- PSRAM について検討する。
 外部に SPI 接続の PSRAM を用意しないといけなさそう。
 IPS6404 というもの。
 Mouser に APS6404L-3SOR-SN(互換)があった。
- camera_pins.h を参考に回路図を作ってみようとするが、 PSRAMのピン割り当てがない。
 また、カメラの書類によってピン番号がかなり異なる。
- 秋月電子で ESP-EYE が 2980 円で販売していた。
 これなら、基板を起こすまでもないか。
 だけど、これだと外部 I/O が使えない。カメラで使っている I2C を
 使えばいいかも。
 でも、ピンがでていない。
- 6. 秋月電子で販売している ESP32-S3-WROOM-1-N16R8 なら 内部に PSRAM を内蔵してさらに USB ポートもあり、 I/O もだいぶ増えている。
 ESP32-WROVER-B も内部に PSRAM を内蔵しているが、 ディスコンになっているし、古い。
- 7. カメラは OV2640 にする。日昇で、600円くらいで販売している。



回路検討

- 1. カメラとのインターフェースのピン番号を変更できるのか。
 - 1. I2C はたぶん問題なし。
 - 2. XCLK, VSYNC, HREF, PCLK は…できるらしい。
 - 3. Y2-Y9 は…できるらしい。
 - 4. RESET は必要か…あればいい。

ESP32S3 Technical Reference Manual \mathcal{O}

Table 62. Peripheral Signals via GPIO Matrix によれば 各入力信号、出力信号は 48本の GPIO ピンに配置できるらしい。 たとえば、CAM_DATA_inO-15 はピン番号を設定できるようです。





i de la de



デバッグ: (2023年5月15日~)

- 1. カメラ用 RESET 信号はつながないと動作しなかった。 (ただし、カメラの種類は M5STACK にて)
- 2. ArduinoIDEの設定で S3 を選択、PSRAMも選択する。

編集 スケッチ ツー				
	自動整形	Ctrl+T		Q
amoraio	スリッナをアーカ1 ノする エンコーディングを修正			
	エノコーフィノンで同止 ライブラリを管理	Ctrl+Shift+I		<u>^</u>
#include	シリアルモニタ	Ctrl+Shift+M		
	シリアルプロッタ	Ctrl+Shift+L		
11	WiFi101 / WiFiNINA Firmware Updater			_
// WARNIN	ESD32 Sketch Data Unload		quality M is selected	_
11	# Kr #EED2362 Dev Medule#		# K77 Sin	
//	Unload Speed: "921600"		Arduino AVR Boards	>
// Soloct	USB Mode: "Hardware CDC and JTAG"	,	ESP32 Arduino	
//#define	USB CDC On Boot: "Disabled"	>	Lor SE Ardanio	 ESP32S3 Dev Module
//#define	USB Firmware MSC On Boot: "Disabled"	>		ESP32C3 Dev Module
#define C	USB DFU On Boot: "Disabled"	>		ESP32S2 Dev Module
//#define	Upload Mode: "UART0 / Hardware CDC"	>	PSRAM	ESP32 Dev Module
//#define	CPU Frequency: "240MHz (WiFi)"	>		ESP32 Wrover Module
//#define	Flash Mode: "QIO 80MHz"	>		ESP32 PICO-D4
//#define	Flash Size: "4MB (32Mb)"	>		ESP32-S3-Box
, adot tild	Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB S	SPIFFS)" >		ESP32-S3-USB-OTG
#include	Core Debug Level: "&L"	>		ESP32S3 CAM LCD
ル 編集 スケッ	チ ツール ヘルプ			
	日則恐形		Ctrl+T	
	日期登形		Ctrl+T	
	目 動 登形 スケッチをアーカイブする		Ctrl+T	
itcameraio	 目 期 登 形 スケッチをアーカイブする エンコーディングを修正 エレコーディングを修正 		Ctrl+T	
atcameraio	 目 動 登 形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 		Ctrl+T Ctrl+Shift+I	
atcameraio 1 #includ 2 #includ	 目 動 登 形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ 		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M	
tcameraio 1 #includ 2 #includ 3	 目 動登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ 		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	
tcameraio 1 #includ 2 #includ 3 4 // 5 // WARN	 日 刻金形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater 		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality
atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 //	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected
atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 0 //	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 //	日 到登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 9 0 // 2010	日刻登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and ITAG"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 0 // Sele 1 // # == 5	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 0 // Sele 1 //#defi 2 //#defi	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 0 // Sele 1 //#defi 2 //#defi	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB Firmware MSC On Boot: "Disabled"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 0 // Sele 1 //#defi 2 //#defi 3 #define	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB Firmware MSC On Boot: "Disabled"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 0 // Sele 1 //#defi 2 //#defi 3 #define 4 //#defi	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" USB DFU On Boot: "Disabled" USB DFU On Boot: "Disabled"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 // 5 // WARN 6 // 7 // 8 // 9 0 // Sele 1 //#defi 2 //#defi 3 #define 4 //#defi 5 //#defi	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" UPload Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 80MHz"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 8 7 7 8 7 7 8 7 7 8 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" ESP32 Sketch Data Upload Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 80MHz" Flash Size: "4MB (32Mb)"		Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 8 7 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルガロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" USB DFU On Boot: "Disabled" Upload Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 800Hz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1	.2MB APP/1.5MB S	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 4 5 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 7 7 7 8 7 7 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルブロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" USB DFU On Boot: "Disabled" Upload Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 800Hz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし"	.2MB APP/1.5MB S	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルガロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" USB DFU On Boot: "Disabled" Upload Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 800Hz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし"	.2MB APP/1.5MB S	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルガロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" USB DFU On Boot: "Disabled" Upload Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 800Hz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし" PSRAM: "OPI PSRAM" Arduino Runs On: "Core 1"	.2MB APP/1.5MB S	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM Disabled QSPI PSRAM
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" EVENTOR (WIFI)" Flash Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "UP S00Hz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし" PSRAM: "OPI PSRAM" Arduino Runs On: "Core 1"	.2MB APP/1.5MB S	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM Disabled QSPI PSRAM
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" EVENTOR (WIFI)" Flash Mode: "QIO 80MHz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし" PSRAM: "OPI PSRAM" Arduino Runs On: "Core 1" Events Run On: "Core 1"	.2MB APP/1.5MB S	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM Disabled QSPI PSRAM
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" EVENT CPU Frequency: "240MHz (WiFi)" Flash Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 80MHz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし" PSRAM: "OPI PSRAM" Arduino Runs On: "Core 1" Events Run On: "Core 1" Erase All Flash Before Sketch Upload: "Disabled	.2MB APP/1.5MB S led"	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM Disabled QSPI PSRAM
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" EVENT CPU Frequency: "240MHz (WiFi)" Flash Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 80MHz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし" PSRAM: "OPI PSRAM" Arduino Runs On: "Core 1" Events Run On: "Core 1" Erase All Flash Before Sketch Upload: "Disabled"	.2MB APP/1.5MB S led"	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM Disabled QSPI PSRAM
Atcameraio 1 #includ 2 #includ 3 4 4 5 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 8 7 7 7 7 7 8 7 7 7 7 7 7 8 7 7 7 7 7 7 7 7 7 7 7 7 7	日朝登形 スケッチをアーカイブする エンコーディングを修正 ライブラリを管理 シリアルモニタ シリアルプロッタ WiFi101 / WiFiNINA Firmware Updater ESP32 Sketch Data Upload ボード: "ESP32S3 Dev Module" Upload Speed: "921600" USB Mode: "Hardware CDC and JTAG" USB CDC On Boot: "Disabled" USB CDC On Boot: "Disabled" USB DFU On Boot: "Disabled" EPU Frequency: "240MHz (WiFi)" Flash Mode: "UARTO / Hardware CDC" CPU Frequency: "240MHz (WiFi)" Flash Mode: "QIO 80MHz" Flash Size: "4MB (32Mb)" Partition Scheme: "Default 4MB with spiffs (1 Core Debug Level: "なし" PSRAM: "OPI PSRAM" Arduino Runs On: "Core 1" Events Run On: "Core 1" Erase All Flash Before Sketch Upload: "Disabled" JTAG Adapter: "Disabled"	.2MB APP/1.5MB S led"	Ctrl+T Ctrl+Shift+I Ctrl+Shift+M Ctrl+Shift+L	quality M is selected buffer size PSRAM Disabled QSPI PSRAM

3. 書き込みと起動

書き込みは USB ポートから可能です。 ただし、モニター出力 (RS232C)は RXD0, TXD0 を使うのでそちらに モニター用の受信ポート (アプリも)を使う。 下記は 書き込みと起動をモニターしたもの。

船 無題 - RSMonitor	-		\times
ファイル(F) 編集(E) 表示(V) ヘルプ(H)			
D 😅 🖬 X 🖻 💼 🚑 💡			
RS-Monitor Program ¥¥.¥COM11 💌 115200 💌 😶 巴巴		CLE	AR
2s3-20210327 Build:Mar 27 2021 rst:0x15 (USB_UART_CHIP_RESET),boot:0x0 (DOWNLOAD(USB/UART0)) Saved PC:0x42082642 waiting for download stild:Mar 27 2021 rst:0x15 (USB_UART_CHIP_RESET),boot:0x8 (SPI_FAST_FLASH_BOOT) Saved PC:0x40378c3e SPIWP:0xee mode:010, clock div:1 load:0x403c8080,len:0x44c load:0x403cc700,len:0x2a38 entry 0x403c9804	- L	小	 ን
 WiFi connected Camera Ready! Use 'http://192.168.0.13' to connect 記動			

SSID とパスワードを書き込んであるので、WiFi に繋がって、カメラが起動。

4. ブラウザで表示させた状態



- 確認のため、CAMERA_MODEL_ESP_EYE を選択して書込みを実行したところ カメラも起動しなければ、その後 ファームの書込みに途中(54%)で 失敗するようになった。
 対処: GPI00 のポートを GND におとし、RXD0, TXD0 のポートを使って 書込みなおしたらもとに戻った。 CAMERA MODEL ESP EYE を使ってはいけない様子。
- 6. HREF が正常でない?

ではなく、これは JPEG データを受けとっている状況でした。 カメラ OV2640 内部で、JPEG 変換をしてそのデータを送っているためでした。 先頭の HREF 以外は HREF と PCLK の AND で、データを吸い上げているものと 思われます。



7. XCLK を変更してみる。

20Mhz...キャプチャーが止まる。安定していない。 30Mhz...キャプチャーが止まらないが画像に筋がはいる。 640X480以下では ヨコシマ、それ以上ではタテシマ 10Mhz...きれいにキャプチャーできる。安定もしている。ただ、少し遅い。



以下は 10Mhz 時の画像(1280 X 1024)

クロックを14Mhz にしたところ、アクセスも安定してきた。 ただ、暗いところ、ヨコシマが目立つような気もす。 以下14Mhz 時の画像(1600 X 1200)



8. ファーム改造方法

ファームを変更するにあたって、確認したことを列記する。

- Html 言語で 192.168.0.13 にアクセスすると、html(javascript を含む)を よみこむ。このhtml データは 圧縮して camera_index.h にはいっている。 圧縮してあるので、修正は不可。 なので、ブラウザで表示させたときに、「検証」モードでソースを確認し、 それをコピーして使用する。
- 2. カメラは 0V2640の他、2機種に対応しているが、そちらは削除しておく。
- htmlのサイズは45kほどあった。これをブラウザでコピーし、それを 送ることにしたところ、下記の画面となった。スタイルがいまいちではあるが 動作には問題なさそう。これで進める。矢印は変更したところ。



45kのhtmlではさすがに多すぎるので html, style, js に分けることに。

```
9. ファーム改造(2023年5月16日)
   App_httpd.cppの中を下記のように、index.html, style.css, index.js を
   アクセスできるようにする。
      httpd uri t index uri = {
          .uri = "/",
          . method = HTTP GET,
          .handler = index_handler,
          .user_ctx = NULL;
      httpd uri t style uri = {
          .uri = "/style.css",
          . method = HTTP GET,
          .handler = style_handler,
          .user_ctx = NULL;
      httpd_uri_t java_uri = {
          .uri = "/index.js",
          . method = HTTP GET,
          .handler = java_handler,
          .user_ctx = NULL;
    (途中省略)
       httpd_register_uri_handler(camera_httpd, &index_uri);
       httpd register uri handler(camera httpd, &style uri);
       httpd_register_uri_handler(camera_httpd, &java_uri);
```

以上、トライしたが、style.css だけは読み込まれても認識しなかったので Index.htmlの中に埋め込むことにした。



いろいろ削除してこんな画面までにはきました。