Cambridge CMOS Sensors

is now

Member of the ams Group

The technical content of this Cambridge CMOS Sensors (CCS) document is still valid.

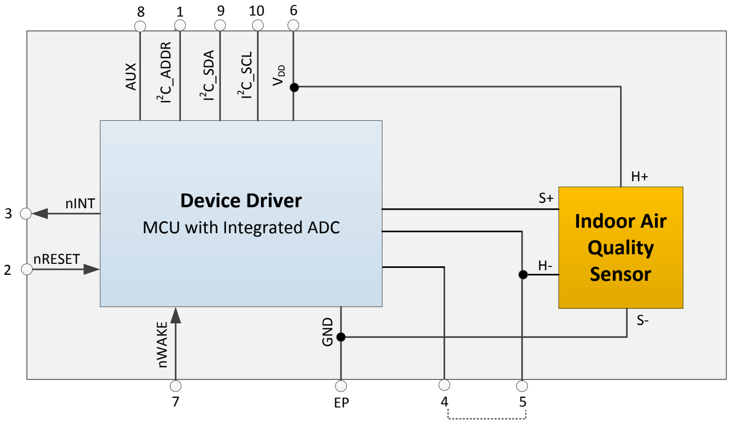
Contact information:

Headquarters:
ams AG
Tobelbader Strasse 30
8141 Premstaetten, Austria
Tel: +43 (0) 3136 500 0
e-Mail: ams_sales@ams.com

Please visit our website at www.ams.com

Ultra-low power digital gas sensor for monitoring indoor air

The CCS811 is an ultra-low power digital gas sensor solution which integrates a metal oxide (MOX) gas sensor for monitoring indoor air quality (IAQ) including a wide range of Volatile Organic Compounds (VOCs) with a microcontroller unit (MCU), a Analog-to-Digital converter (ADC), and an I^{2}C interface.



CCS811 Block Diagram

CCS811 is based on Cambridge CMOS sensors unique Micro-hotplate technology enables a highly reliable solution for gas sensors, very fast cycle times and a significant reduction in average power consumption vs. traditional MOX gas sensors.

The integrated MCU manages the sensor drive modes, ADC measurements and raw sensor data measured while detecting VOCs. The I^{2}C digital interface significantly simplifies the hardware and software design, enabling a faster time to market.

CCS811 supports intelligent detection algorithms to process raw sensor measurements to represent equivalent CO_2 (eCO_2) levels or TVOC measurement in real world environments, where the main cause of VOCs is from humans.

CCS811 supports multiple measurement modes that have been optimised for low-power consumption during an active sensor measurement and idle mode extending battery life in portable applications.

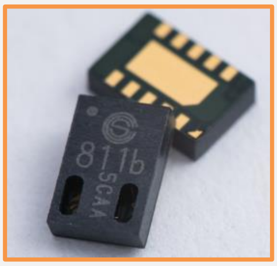
CCS811 is available in a 10 lead 2.7 x 4.0mm, 0.6mm pitch LGA package.

Features

- IAQ Gas sensor
- Integrated MCU with ADC
- I^{2}C digital interface
- Optimised low-power modes
- Automatic and manual baseline correction
- 2.7x4.0 mm LGA package
- Proven technology platform

Benefits

- On-board processing to reduce requirement on host processor
- Fast time-to-market
- Extended battery life
- Reduced component count
- Suitable for small form factor designs
- Highly reliable solution



Applications

- Smartphones
- Wearables
- Smart Home devices
- Accessories

Electrical characteristics

Parameters	Conditions	Min	Typ	Max	Units
Recommended operating temperature		-5		+50	°C
Storage temperature		-40		+125	°C
Supply Voltage (V_{DD})		1.8		3.6	V
Supply Current (I_{DD}) [1]	During measuring		26		mA
	Average over pulse cycle [2]		0.7		mA
	Sleep Mode		19		µA
Power Consumption	Idle Mode 0 at V_{DD}= 1.8V		34		µW
	Mode 1 & 4 at V_{DD}= 1.8V		46		mW
	Mode 2 at V_{DD}= 1.8V		7		mW
	Mode 3 at V_{DD}= 1.8V		1.2		mW
Logic High Input[3]	nRESET, nWAKE, I^2C_ADDR	V_{DD} - 0.6		V_{DD}	V
Logic Low Input[3]	nRESET, nWAKE, I^2C_ADDR	0		0.6	V
Logic High Output	nINT	V_{DD} - 0.7			V
Logic Low Output	nINT			0.6	V
Analogue Input	AUX	0		V_{DD}	V

Notes:

1. Typical values for 1.8V supply voltage (V_{DD})
2. Average Supply Current (I_{DD}) for a sensor measurement once every 60 seconds
3. For SDA and SCL timing refer NXP I^2C – bus specification and user manual UM10204

Table 1: Electrical Characteristics

Timing characteristics

Parameters	Conditions	Min	Typ	Max	Units
T_{AWAKE}[1]	Time until active after nWAKE asserted.	50			µs
T_{DWAKE}	Minimum time nWAKE should be de-asserted	20			µs
T_{RESET}	Minimum nRESET low pulse	20			µs
T_{START}[2]	Time until active after Power on		18	20	ms
	Time until active after nRESET		1	2	ms
F_I^2C	Frequency of I^2C Bus Supported	10	100	400	KHz

Notes:

1. nWAKE should be asserted prior to and during any I^2C transaction
2. Up to 70ms on the first Reset after new application download

Table 2: Timing Characteristics

CCS811 Operation

Modes of operation

The CCS811 has 5 modes of operation as follows

- Mode 0: Idle, low current mode
- Mode 1: Constant power mode, IAQ measurement every second
- Mode 2: Pulse heating mode IAQ measurement every 10 seconds
- Mode 3: Low power pulse heating mode IAQ measurement every 60 seconds
- Mode 4: Constant power mode, sensor measurement every 250ms

In Modes 1, 2, 3, the equivalent CO_2 concentration (ppm) and TVOC concentration (ppb) are calculated for every sample.

- Mode 1 reacts fastest to gas presence, but has a higher operating current
- Mode 3 reacts more slowly to gas presence but has the lowest average operating current.

It is not possible to change drive modes on the fly in all cases. When a sensor operating mode is changed to a new mode with a lower sample rate (e.g. from Mode 1 to Mode 3), it should be placed in Mode 0 (Idle) for at least 10 minutes before enabling the new mode. When a sensor operating mode is changed to a new mode with a higher sample rate (e.g. from Mode 3 to Mode 1), there is no requirement to wait before enabling the new mode.

Mode 4 is intended for systems where an external host system wants to run an algorithm with raw data (e.g. Alcohol breathalyser application) and this mode provides new sample data every 250ms.

Note: Mode timings are subject to typical 2% tolerance due to accuracy of internal clock

Early-Life Use

CCS811 performance in terms of resistance levels and sensitivities will change during early life use. During this phase the change in resistance is greatest over the first 24 – 48hrs of operation. Burn-in is a process to accelerate this phase of the CCS811, and thereby minimise changes during normal operation.

CCS recommends where possible CCS811 runs in continuously for at least 24hrs in the selected mode 1 – 4 to ensure performance is more stable and can be used to provide a more accurate indication of the IAQ level.

TVOC

The Total Volatile Organic Compound (TVOC) detection range for CCS811 is from 0ppb to 1000ppb
This is calibrated to a typical TVOC mixture in an indoor environment. If the ratio of compounds in the environment is significantly different the TVOC output will be affected as some VOC compounds will have greater or lesser effect on the sensor.

eCO_2

The equivalent CO2 (eCO_2) detection range for CCS811 is from 400ppm to 5000ppm.

Temperature and Humidity Compensation

If an external sensor is available this information can be written to CCS811 and used to compensate for temperature and humidity changes. *Refer to the ENV_Data Register.* If an external temperature sensor is not available the external NTC circuit can be used to determine the ambient temperature and write this information to CCS811.

Interrupt and Interrupt on threshold

At the end of each measurement cycle (250ms, 1s, 10s, 60s) a flag is set and optionally interrupt (nINT) pin asserted. *Refer to the MEAS_ Mode Register.* The user can choose to only assert nINT if the eCO2 value changes into a different range set by register values. *Refer to the Thresholds register.*

Optional NTC Circuit (AUX)

The NTC Register provides the voltages across the R_{NTC} and the R_{REF} if fitted (mV) – to enable host system to calculate the ambient temperature. The calculated temperature can be written to CCS811 to compensate for temperature changes.

Automatic Baseline Correction

The CCS811 continuously monitors the baseline that is used to calculate the TVOC and eCO_2 concentrations in modes 1-3. Gas concentrations are expected to vary in a typical environment so the minimum time over which a baseline correction is calculated is 24 hours.

Corrections are expected to be small but may occasionally be observed.

Manual Baseline correction

There are 2 methods available to correct the baseline manually.

- If the host system knows that it is in clean air it can reset the baseline to the current value by re-writing the current mode (e.g. Mode 1, 2 and 3) to the MEAS_MODE register

- If a previous encoded Baseline value has been saved from the BASELINE register (when the air was known to be clean) it can be written back to the BASELINE register by the host system[1]

Note:

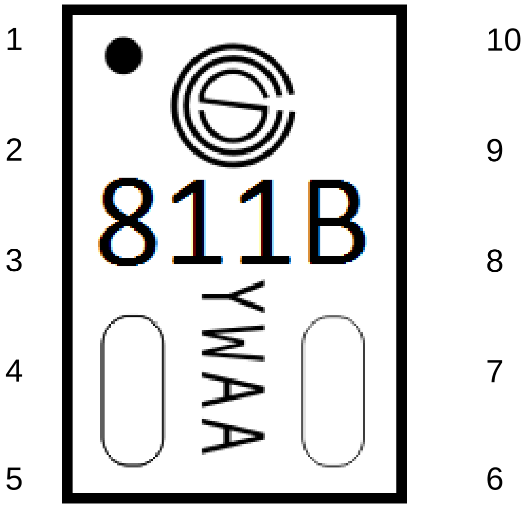
1. If a value is written to the BASELINE register while the sensor is stabilising, the output of the TVOC and eCO_2 calculations may be higher than expected.

For additional information on using the CCS811 please refer to application note CC-000803-AN: CCS811 Programming and Interfacing Guide.

Pin assignment

Pin No	Name	Description
1	I^{2}C _ADDR	Single address select bit to allow alternate address to be selected • When I^{2}C _ADDR is low the 7 bit I^{2}C address is decimal 90 / hex 0x5A • When I^{2}C _ADDR is high the 7 bit I^{2}C address is decimal 91 / hex 0x5B
2	nRESET	nRESET is an active low input and is pulled up to V$_{DD}$ by default. nRESET is optional but external 4.7KΩ pull-up and/or decoupling of the nRESET pin may be necessary to avoid erroneous noise-induced resets.
3	nINT	nINT is an active low optional output.. It is pulled low by the CCS811 to indicate end of measurement or a set threshold value has been triggered
4	PWM	Heater Driver PWM output. Connect pins 4 and 5 together
5	Sense	Heater current sense. Connect pins 4 and 5 together
6	V$_{DD}$	Supply voltage
7	nWAKE	nWAKE is an active low input and should be asserted by the host prior to an I^{2}C transaction and held low throughout.
8	AUX	Optional AUX pin which can be used for ambient temperature sensing with an external NTC resistor. If not used leave unconnected.
9	I^{2}C _SDA	I^{2}C _SDA pin is used for I^{2}C data. Should be pulled up to V$_{DD}$ with a resistor
10	I^{2}C _SCK	I^{2}C _SCK pin is used for I^{2}C clock. Should be pulled up to V$_{DD}$ with a resistor
EP	Exposed Pad	Connect to ground

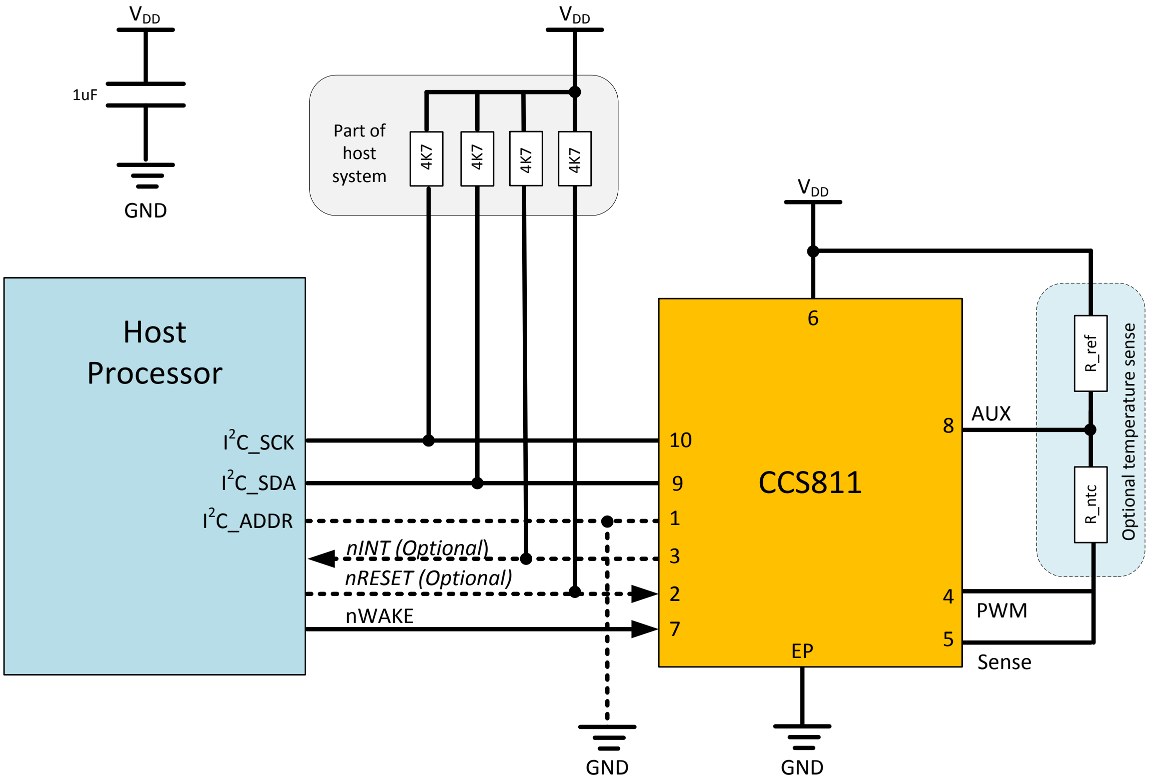
Table 3: CCS811 LGA Pin Assignment



The Exposed Pad is underneath (see Table)

Recommended application circuit

The recommended application circuit for CCS811 is shown below:



Notes:

1. Pull-up resistors for I^{2}C SCK and I^{2}C SDA assumed will be part of host system
2. AUX Pin on CCS811 can be used for optional ambient temperature sensing, if not used leave unconnected.

Host system software requirements

1. The minimum level of driver support that a host system needs is read and write I^{2}C transactions where the nWAKE pin is asserted at least 50µs before the transaction and kept asserted throughout

   - I^{2}C Write transactions of 1-N data bytes
   - I^{2}C Read transactions of 1-N data bytes

2. An Interrupt handler is also recommended to tell the application code that the device has asserted an interrupt.

I²C interface

I²C transactions require a register address to be selected (written) and followed by data, as described by the transaction types in the subsections below. Each Register location corresponds to a byte or multiple bytes (like a FIFO). The number of bytes for a given Register can be different for different algorithms.

Multiple reads or writes in a single sequence will be to or from the same location (the address does not increment).

I²C Register Write

Write, register address only (set or action)	S	SLAVE ADDRESS	W	A	DATA(REG ADDR)	A	P				
Write value to register	S	SLAVE ADDRESS	W	A	DATA(REG ADDR)	A	DATA	A	P		
Write multiple data values to register (letterbox)	S	SLAVE ADDRESS	W	A	DATA(REG ADDR)	A	DATA	A	DATA	A	P

As shown above, a transaction may be:

- Single Data Byte to select a register address for subsequent read
- Two Data Bytes to select a register address and write a byte to it, typically to set a single-byte register value
- Multi-Bytes to select a register address and write several bytes to it, typically to set multiple configuration bytes

I²C Register Read

Select and read register value	S	SLAVE ADDRESS	W	A	DATA(REG ADDR)	A	P	(select register address)						
	S	SLAVE ADDRESS	R	A	DATA	N	P	(read register value)						
Select and read register value, as single transaction	S	SLAVE ADDRESS	W	A	DATA(REG ADDR)	A	R	SLAVE ADDRESS	R	A	DATA	N	P	
	(select register address)							(read register value)						
Select and repeatedly read multiple data values from register	S	SLAVE ADDRESS	W	A	DATA(REG ADDR)	A	P	(select register address)						
	S	SLAVE ADDRESS	R	A	DATA	A	DATA	N	P	(read register multiple data)				
	S	SLAVE ADDRESS	R	A	DATA	A	DATA	N	P	(read register multiple data)				
	S	SLAVE ADDRESS	R	A	DATA	A	DATA	N	P	(read register multiple data)				

Since no register address can be supplied during an I²C read, an I²C write to select the required register first. The write and read operations can optionally be combined into a single transaction using a repeated start condition, as shown in the second example above. Reads may then be single or multi-byte, dependent on the data stored at that address.

Application Register map

All I^{2}C transactions must use this address 0x5A or 0x5B depending on status of I^{2}C _ADDR pin when writing to and reading from the CCS811. **Table 4** below shows the register map for CCS811. **Table 17** shows the bootloader register map.

Address	Register	R/W	Size	Description
0x00	STATUS	R	1 byte	Status register
0x01	MEAS_MODE	R/W	1 byte	Measurement mode and conditions register
0x02	ALG_RESULT_DATA	R	up to 8 bytes	Algorithm result. The most significant 2 bytes contain a ppm estimate of the equivalent CO_2 (eCO_2) level, the least significant 2 bytes contain a ppb estimate of the total VOC level.
0x03	RAW_DATA	R	2 bytes	Raw ADC data values for resistance and current source used. The most significant 6-bits store the selected current source for the reading. The least significant 10 bits store the raw ADC value.
0x05	ENV_DATA	W	4 bytes	Temperature and Humidity data can be written to enable compensation
0x06	NTC	R	4 bytes	Provides the voltage across the reference resistor and the voltage across the NTC resistor – from which the ambient temperature can be determined.
0x10	THRESHOLDS	W	5 bytes	Thresholds for operation when interrupts are only generated when eCO2 ppm crosses a threshold
0x11	BASELINE	R/W	2 bytes	The encoded current baseline value can be read. A previously saved encoded baseline can be written.
0x20	HW_ID	R	1 byte	Hardware ID. The value is 0x81
0x21	HW Version	R	1 byte	Hardware Version. The value is 0x1X
0x23	FW_Boot_Version	R	2 bytes	Firmware Boot Version. The first 2 bytes contain the firmware version number for the boot code.
0x24	FW_App_Version	R	2 bytes	Firmware Application Version. The first 2 bytes contain the firmware version number for the application code
0xE0	ERROR_ID	R	1 byte	Error ID. When the status register reports an error it source is located in this register
0xFF	SW_RESET	W	4 bytes	If the correct 4 bytes (0x11 0xE5 0x72 0x8A) are written to this register in a single sequence the device will reset and return to BOOT mode.

Table 4: CCS811 Application Register Map

For more information on CCS811 programming requirements please refer to application note CC-000803-AN

STATUS Register (0x00)

Single byte read only register which indicates if a device is active, if a new data is available or if an error occurred.

Table 5 below shows the bit fields contained in the status register.

7	6	5	4	3	2	1	0
FW_MODE	-		APP_VALID	DATA_READY	-		ERROR

Table 5: Status Register

Table 6 below describes the bit fields contained in the status register.

Bit(s)	Field	Description
7	FW_MODE	0: Firmware is in boot mode, this allows new firmware to be loaded 1: Firmware is in application mode. CCS811 is ready to take ADC measurements
6:5	-	Reserved
4	APP_VALID	0: No application firmware loaded 1: Valid application firmware loaded
3	DATA_READY	0: No new data samples are ready 1: A new data sample is ready in ALG_RESULT_DATA, this bit is cleared when ALG_RESULT_DATA is read on the I^{2}C interface
2:1	-	Reserved
0	ERROR	0: No error has occurred on I^{2}C or the sensor 1: There is an error on the I^{2}C or sensor, the ERROR_ID register (0xE0) contains the error source

Table 6: Status Register Fields

MEAS_MODE (Measurement and Conditions) Register (0x01)

This is Single byte register, which is used to enable sensor drive mode and interrupts.

Table 7 below shows the bit fields contained in the measure mode register.

7	6:4	3	2	1	0
-	DRIVE_MODE	INTERRUPT	THRESH	-	

Table 7: Measure Mode Register

Table 8 below describes the bit fields contained in the Measure Mode register.

Bit(s)	Field	Description
7	-	Reserved – write '0'
6:4	DRIVE_MODE	000: Mode 0 – Idle (Measurements are disabled in this mode) 001: Mode 1 – Constant power mode, IAQ measurement every second 010: Mode 2 – Pulse heating mode IAQ measurement every 10 secs 011: Mode 3 – Low power pulse heating mode IAQ measurement every 60 secs 100: Mode 4 – Constant power mode, sensor measurement every 250ms 1xx : Reserved modes (For future CCS use) In mode 4 raw data mode a measurement is performed every 250ms for external algorithms. A new sample is placed in ALG_RESULT_DATA and RAW_DATA registers and the DATA_READY bit in the STSTUS register is set at the defined measurement interval.
3	INTERRUPT	0: Interrupt generation is disabled 1: The nINT signal is asserted (driven low) when a new sample is ready in ALG_RESULT_DATA. The nINT signal will stop being driven low when ALG_RESULT_DATA is read on the I^2C interface. At the end of each measurement cycle (250ms, 1s, 10s, 60s) a flag is set in the STATUS register regardless of the setting of this bit..
2	THRESH	0: Interrupt mode (if selected) operates normally 1: Interrupt mode (if selected) only asserts the nINT signal if the new ALG_RESULT_DATA crosses one of the thresholds set in the THRESHOLDS register (0x10) by more than the hysteresis value (also in the THRESHOLDS register)
1:0	-	Reserved

Table 8: Measure Mode Register Fields

ALG_RESULT_DATA (Algorithm Results Data) Register (0x02)

This multi-byte read only register contains the calculated eCO_2 (ppm) and TVOC (ppb) values followed by the STATUS register, ERROR_ID register and the RAW_DATA register.

- If only eCO_2 is required, only the first 2 bytes need to be read.
- If TVOC is required, 4 bytes need to be read.
- In a system where interrupts are not implemented and the host needs to poll the STATUS register to determine whether there is new data, an efficient alternative is to read 5 bytes in a single transaction and that returns eCO_2, TVOC and the status register.
- Optionally, all 8 bytes could be read in a single transaction

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6 & 7
eCO_2 High Byte	eCO_2 Low Byte	TVOC High Byte	TVOC Low Byte	STATUS	ERROR_ID	See RAW_DATA

Table 9: Algorithm Results Register Byte Order

RAW_DATA Register (0x03)

Two byte read only register which contains the latest readings from the sense resistor.
The most significant 6 bits of the first byte contains the value of the current through the sensor (1 to 63uA)
The lower 10 bits contains the raw ADC reading of the voltage across the sensor with the selected current (1023 = 1.65V)

Byte 0								Byte 1							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Current Selected 5:0						Raw ADC reading 9:0									

Table 10: Raw Data Register Byte Order

ENV_DATA (Environment Data) Register (0x05)

A multi-byte register that can be written with the current Humidity and Temperature values if known.

Byte 0								Byte 1								Byte 2								Byte 3							
Humidity High Byte								Humidity Low Byte								Temperature High Byte								Temperature Low Byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Humidity %							Humidity % Fraction									Temperature + 25°C							Temperature + 25°C Fraction								

Table 11: Environment Register Fields and Byte Order

Humidity is stored to the nearest 0.5% in byte 0, additional fractions can be stored in byte 1. The default value is 50% = 0x64, 0x00. As an example 48.5% humidity would be 0x61, 0x00.

Temperature is stored to the nearest 0.5°C in byte 2 with 25°C added as an offset (so all values are positive), additional fractions can be stored in byte 3. The default value is 25°C = 0x64, 0x00. As an example 23.5% temperature would be 0x61, 0x00.

The internal algorithm uses these values (or default values if not set by the application) to compensate for changes in humidity and ambient temperature.

NTC Register (0x06)

Four byte read only register which contains the voltage across resistor and the voltage across the NTC resistor from which the ambient temperature can be determined.

The resistance of the NTC resistor can be determined by the ratio value and knowledge of the reference resistor used.

$R_{NTC} = V_{NTC} \times R_{REF} / V_{REF}$. The temperature can be determined from the resistance of the NTC resistor from the datasheet of the NTC resistor used.

Byte 0	Byte 1	Byte 2	Byte 3
Voltage across R_{REF} (mV)		Voltage across R_{NTC} (mV)	
High Byte	Low Byte	High Byte	Low Byte

Table 12: NTC Register Byte Order

This enables the host to calculate the ambient temperature and this information can be written to CCS811 to compensate for temperature changes.

THRESHOLDS Register (0x10)

If 'interrupt on threshold change' has been set in the Mode register (see above), the values in this multi- byte write only register are used to determine the thresholds and the level of hysteresis desired.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
Low to Medium Threshold		Medium to High Threshold		Hysteresis Value
High Byte	Low Byte	High Byte	Low Byte	

Table 13: Thresholds Register Byte Order

An interrupt is asserted if the eCO_2 value moved from the current range (Low, Medium, or High) into another range by more than the Hysteresis value (used to prevent multiple interrupts close to a threshold).

- Low to Medium Threshold default = 1500ppm = 0x05DC
- Medium to High Threshold default = 2500ppm = 0x09C4
- Hysteresis value default = 50 = 0x32

BASELINE Register (0x11)

A two byte read/write register which contains an encoded version of the current baseline used in Algorithm Calculations.

A previously stored value may be written back to this two byte register and the Algorithms will use the new value in its calculations (until it adjusts it as part of its internal Automatic Baseline Correction).

If a baseline is to be saved and later restored, both bytes must be saved together and written back together.

HW_ID (Hardware identifier) Register (0x20)

Single byte read only register which holds the HW ID which is 0x81 for this family of CCS81x devices.

HW_ Version (Hardware Version) Register (0x21)

Single byte read only register which holds the Hardware Major and Minor Hardware versions.
The top four bits read major hardware version 1 – identifying the product as CCS811. The bottom four bits identify any build variant. The default value is 0x1X.

FW_Boot_Version (Firmware Bootloader Version) Register (0x23)

Two byte read only register which contain the version of the firmware bootloader stored in the CCS811 in the format Major.Minor.Trivial

Byte 0								Byte 1							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Major				Minor				Trivial							

Table 14: Firmware Bootloader Version Format

FW_App_Version (Firmware Application Version) Register (0x24)

Two byte read only register which contain the version of the firmware application stored in the CCS811 in the format Major.Minor.Trivial

Byte 0								Byte 1							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Major				Minor				Trivial							

Table 15: Firmware Application Version Format

ERROR_ID (Error Identifier) Register (0xE0)

If the ERR bit [0] of the STATUS Register is set, this single byte read only register indicates source(s) of the error.

Table 16 describes the bit fields contained in the Error ID Register.

Bit	ERROR_CODE	Description
0	MSG_INVALID	The CCS811 received an I^2C write request addressed to this station but with invalid mailbox ID
1	READ_REG_INVALID	The CCS811 received an I^2C read request to a mailbox ID that is invalid
2	MEASMODE_INVALID	The CCS811 received an I^2C request to write an unsupported mode to MEAS_MODE
3	MAX_RESISTANCE	The sensor resistance measurement has reached or exceeded the maximum range
4	HEATER_FAULT	The Heater current in the CCS811 is not in range
5	HEATER_SUPPLY	The Heater voltage is not being applied correctly
6	-	Reserved for Future Use
7	-	Reserved for Future Use

Table 16: Error ID Register Codes

SW_RESET Register (0xFF)

As an alternative to Power-On reset or Hardware Reset a Software Reset is available.

Asserting the SW_RESET will restart the CCS811 in Boot mode to enable new application firmware to be downloaded or simply a clean restart of the CCS811.

To prevent accidental SW_RESET a sequence of four bytes must be written to this register in a single I^2C sequence: 0x11, 0xE5, 0x72, 0x8A.

For details, please refer to application notes CC-000803-AN and CC-000922-AN.

Bootloader Register map

All I^{2}C transactions must use this address 0x5A or 0x5B depending on status of I^{2}C _ADDR pin when writing to and reading from the CCS811. **Table 17** below shows the register map for CCS811.

Address	Register	R/W	Size	Description
0x00	STATUS	R	1 byte	Status register
0x20	HW_ID	R	1 byte	Hardware ID. The value is 0x81
0x21	HW Version	R	1 byte	Hardware Version. The value is 0x1x
0x23	FW_Boot_Version	R	2 bytes	Firmware Boot Version. The first 2 bytes contain the firmware version number for the boot code.
0x24	FW_App_Version	R	2 bytes	Firmware Application Version. The first 2 bytes contain the firmware version number for the application code.
0xE0	ERROR_ID	R	1 byte	Error ID. When the status register reports an error it source is located in this register
0xF1	APP_ERASE	W	4 bytes	If the correct 4 bytes (0xE7 0xA7 0xE6 0x09) are written to this register in a single sequence the device will start the application erase
0xF2	APP_DATA	W	9 bytes	Transmit flash code for the bootloader to write to the application flash code space.
0xF3	APP_VERIFY	W	-	Starts the process of the bootloader checking though the application to make sure a full image is valid.
0xF4	APP_START	W	-	Application start. Used to transition the CCS811 state from boot to application mode, a write with no data is required. Before performing a write to APP_START the Status register should be accessed to check if there is a valid application present.
0xFF	SW_RESET	W	4 bytes	If the correct 4 bytes (0x11 0xE5 0x72 0x8A) are written to this register in a single sequence the device will reset and return to BOOT mode.

Table 18: CCS811 Bootloader Register Map

Notes:

1. *For more information on* performing application code download *please refer to application note CC-000922-AN*

Registers not detailed below are documented in the Application Register Map section above.

APP_ERASE (Application Erase) Register (0xF1)

To prevent accidental APP_ERASE a sequence of four bytes must be written to this register in a single I^{2}C sequence: 0xE7, 0xA7, 0xE6, 0x09.

The APP_ERASE can take a variable amount of time. The status register can be polled to determine when this function is complete. The 6th bit (0x40) is initialised to 0 and set to a 1 on completion of the APP_ERASE function. After an erase this bit is only cleared by doing a reset or starting the application.

APP_DATA (Application Data) Register (0xF2)

Nine byte, write only register for sending small chunks of application data which will be written in order to the CCS811 flash code.

The top four bits read major hardware version 1 – identifying the product as CCS811. The bottom four bits identify the build variant. The default value is 0x1x.

APP_VERIFY (Application Verify) Register (0xF3)

Single byte write only register which starts the application verify process run by the bootloader to check for a complete application code image. Command only needs to be called once after a firmware download as the result is saved in a flash location that gets checked during device initialisation.

The APP_VERIFY can take a variable amount of time. The status register can be polled to determine when this function is complete. The 5th bit (0x20) is initialised to 0 and set to a 1 on completion of the APP_VERIFY function. After an APP_VERIFY this bit is only cleared by doing a reset or starting the application.

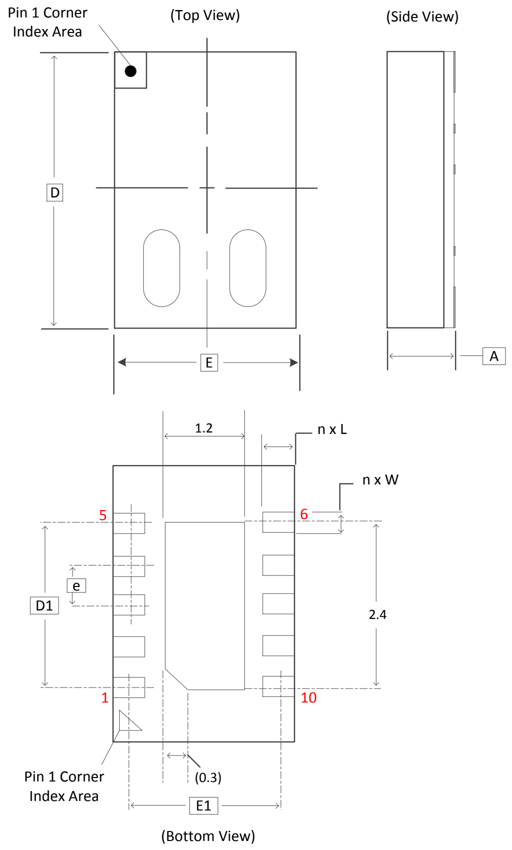
For details on downloading new application firmware please refer to application notes CC-000803-AN and CC-000922-AN.

APP_START (Application Start) Register (0xF4)

To change the mode of the CCS811 from Boot mode to running the application, a single byte write of 0xF4 is required.

The CCS811 interprets this as an address write to select the 'APP_START' register and starts running the loaded application software if it is a valid version (*Refer to the Status Register*).

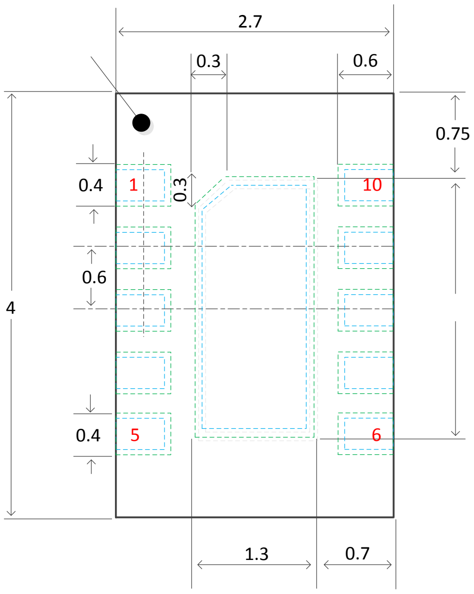
LGA package outline



	Symbol	Dimensions		
		Min	Nominal	Max
Total thickness	A	-	-	1.1
Body Size	D		4.0	BSC
	E		2.7	BSC
Lead Width	W	0.25	0.3	0.35
Lead Length	L	0.45	0.5	0.55
Lead Pitch	e		0.6	BSC
Lead Count	n		10	
Edge Lead Centre to Centre	D1		2.4	BSC
	E1		2.2	BSC
Note: All dimensions are in mm				

Table 18: LGA Package Dimensions

The recommended package footprint or landing pattern for CCS811 is shown below:



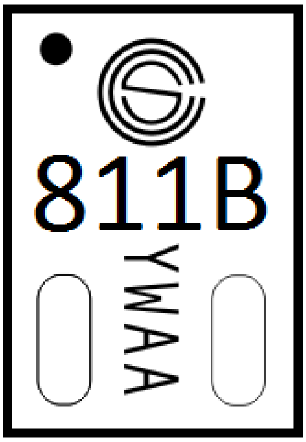
(Top View)

Note:

1. All dimensions are in mm
2. PCB land pattern in Green dash lines
3. Pin numbers are in Red
4. Add 0.05mm all around the nominal lead width and length for the PCB land pattern

Product marking

Product marking for CCS811 is shown below:



Notes:

1. Product code is 811B
2. YWAA indicates the manufacture trace code
   a. Y = Year (2015=5, 2016 = 6)
   b. W = Work Week (01=a, 26=z, 27=A,52=Z)
   c. AA-Lot sequence (1st = AA, 2nd = AB…)

References

Document Reference	Description
CC-000774-AN	Assembly guidelines for CCS811
CC-000783-AN	Mechanical considerations for CCS811
CC-000803-AN	CCS811 programming and interfacing guide
CC-000921-AN	CCS811 Clean Air Baseline Save and Restore
CC-000922-AN	CCS811 Performing a Firmware download
CC-000925-AN	Connecting an NTC Thermistor to the CCS811
CC-000985-AN	CCS811 Manufacture Test Procedure
CC-001011-WP	IAQ User experience during initial operation

Ordering information

Part Number	Description	Package[1,2]	MOQ
CCS811B-JOPR	CCS811 digital gas sensor for Indoor Air Quality Monitoring	2.7 x 4.0mm LGA	5K
CCS811B-JOPS	Sample of CCS811 digital gas sensor for Indoor Air Quality Monitoring	2.7 x 4.0mm LGA	100

Note:

1. Refer to JEDEC J-STD020 lead-free standard for typical soldering reflow profile

The contents of this document are subject to change without notice. While every care has been taken to ensure the accuracy of the contents of this document, CCS cannot accept responsibility for any errors. CCS products are not designed, authorized or warranted for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of a CCS product can reasonably be expected to result in personal injury, death or severe property or environmental damage. CCS accepts no liability for inclusion and/or use of CCS products in such equipment or applications and therefore such inclusion and/or use, is at the customer's own risk. As any devices operated at high temperature have inherently a certain rate of failure, it is therefore necessary to protect against injury, damage or loss from such failures by incorporating appropriate safety measures. CCS products are not designed to work in condensing high humidity environments