

SEZNAM PŘÍLOH

A	Audio kodek api – referenční manuál	I
A.1	Pin	I
A.2	I ² C	I
A.3	I ² S	II
A.4	Audio kodek	II
A.5	Ovládání hlasitosti	XII
B	Schéma zapojení	XIV
C	Seznam součástek	XIX
D	DPS	XXI
E	Obsah CD	XXIII

A AUDIO KODEK API – REFERENČNÍ MANUÁL

Knihovna pro práci s audio kodekem TLV320AIC3101 je napsána tak, aby byla platformě nezávislá. Toho je dosaženo využitím několika tříd, poskytujících jednotné rozhraní k hardwaru. Jsou to třídy `pin_t`, `i2c_t` a `i2s_t`.

A.1 Pin

Třída `pin_t` umožňuje nezávislé ovládání jednotlivých pinů mikroprocesoru. Musí na ní být definovány následující datové typy:

- `enum peripheral_select { PA, PB };`

A následující metody:

- `void make_peripheral(const peripheral_select& peripheral)`
Nastaví, která z periférií mikroprocesoru je k pinu připojena. Využívá `i2c_t` a `i2s_t`.
- `void make_low()`
Nastaví pin jako výstupní s hodnotou log. 0.
- `void set_low()`
Nastaví na pinu log. 0.
- `void set_high()`
Nastaví na pinu log. 1.

A.2 I²C

Komunikace s audio kodekem probíhá přes I²C linku. Tu zapouzdřuje třída `i2c_t`, na které musí být definovány následující metody:

- `bool write(const uint8_t& dev_addr, const uint8_t& reg_addr, const uint8_t& data)`
- `bool write(const uint8_t& dev_addr, const uint8_t& reg_addr, const uint16_t& data)`
Zapíše data do registru.

Parametry:

- `dev_addr` – fyzická adresa zařízení na sběrnici
- `reg_addr` – adresa registru, do kterého se má zapsat
- `data` – data, která se mají do registru zapsat

Návratová hodnota: `true` pokud funkce uspěla, jinak `false`.

- `int read(const uint8_t& dev_addr, const uint8_t& reg_addr, const uint8_t&)`
- `int read(const uint8_t& dev_addr, const uint8_t& reg_addr, const uint16_t&)`
Přečte data z registru.

Parametry:

- `dev_addr` – fyzická adresa zařízení na sběrnici
- `reg_addr` – adresa registru, ze kterého se má číst

- Prázdný parametr, sloužící pouze pro určení velikosti čtených dat
- Návratová hodnota:** Přečtená hodnota pokud funkce uspěla, jinak -1.

Vzhledem k tomu, že komunikace po I²C sběrnici se nemusí vždy podařit, je vhodné, aby funkce měly interní timeout a nedocházelo k zamrzání aplikace.

A.3 I²S

Třída `i2s_t` zajišťuje rozhraní pro přenos digitálního audio signálu mezi kodekem a mikroprocesorem. Musí na ní být definovány následující metody:

- `bool empty()`
Vrátí `true`, pokud nejsou k dispozici nová data z kodeku, jinak `false`.
- `uint32_t read()`
Vrátí naposled přijatá data z kodeku.
- `void write(const uint32_t& data)`
Zapíše data do kodeku.

A.4 Audio kodek

Audio kodek je reprezentován třídou `audio_codec_t`. Na ní jsou definované následující datové typy:

- `enum sample_rate_t`
 - `F48kHz` = 0
 - `F32kHz` = 1
 - `F24kHz` = 2
 - `F19200Hz` = 3
 - `F16kHz` = 4
 - `F12kHz` = 6
 - `F10667Hz` = 7
 - `F9600Hz` = 8
 - `F8kHz` = 10
 - `F96kHz` = 16 + 0
 - `F64kHz` = 16 + 1
 - `F38400Hz` = 16 + 3

Nastavení vzorkovací frekvence převodníků (pouze pokud $f_{S(\text{ref})}$ je 48 kHz).

- `enum mic_bias_t`
 - `MICBIAS_DISABLED` = 0
 - `MICBIAS_2V` = 1
 - `MICBIAS_2V5` = 2
 - `MICBIAS_AVDD` = 3
- `enum stereo_output_config_t`
 - `SINGE_ENDED` = 0
 - `PSEUDO_DIFF` = 1
 - `FULLY_DIFF` = 2
- `enum fsref_t`
 - `FSref48kHz` = 0
 - `FSref44_1kHz` = 1
- `enum audio_interface_mode_t`

- I2S = 0
- DSP = 1
- LEFT_JUSTIFIED = 2
- RIGHT_JUSTIFIED = 3
- enum audio_interface_word_length_t
 - W16BITS = 0
 - W20BITS = 1
 - W24BITS = 2
 - W32BITS = 3

U většiny metod jejich název, případně jména a typy parametrů, dostatečně popisují jejich funkce. Proto je zde, až na výjimky, uveden pouze seznam metod.

Nastavovací metody vrací `true`, pokud uspějí, v opačném případě `false` (viz A.2 I2C). Vyčítající metody vrací hodnoty ve stejném rozsahu, jako korespondující nastavovací metody. Výjimkou je, pouze pokud selžou. V tom případě vrací `-1`.

Definované metody:

- `audio_codec_t (i2c_t &contorl_interface, i2s_t data_interface, pin_t rst_pin)`
Konstruktor.
- `bool init (const sample_rate_t& sample_rate = F48kHz, const fsref_t f& fsref = FSref48kHz, const audio_interface_word_length_t& word_len = W16BITS, const clock_t::clk_in_source_t& clock_in = clock_t::MCLK, const bool& bclk_master = true, const bool& wclk_master = true, const uint8_t& pll_J = 8, const uint16_t& pll_D = 1920, const uint8_t& pll_R = 1, const uint8_t& pll_P = 1, const high_power_output_stage_t::common_mode_voltage_t& common_mode_voltage = high_power_output_stage_t::CMV_1V65, const bool& left_analog_mic = true, const bool& right_analog_mic = true)`
- `bool set_sample_rate (const sample_rate_t& sample_rate)`
- `int get_sample_rate ()`
- `bool set_FSref (const fsref_t& fsref)`
- `int get_FSref ()`
- `bool enable_3D_effect (const bool& enable = true)`
- `bool disable_3D_effect ()`
- `int is_3D_effect_enabled ()`
- `bool set_mic_bias (const mic_bias_t& mic_bias)`
- `int get_mic_bias ()`
- `int is_headset_detected ()`
- `int was_headset_status_changed ()`
- `bool mute_while_resync (const bool& en = true)`
- `bool dont_mute_while_resync ()`
- `int is_muted_while_resync ()`
- `bool set_capless_driver ()`
- `bool set_ac_coupled_driver ()`

- `int is_ac_coupled_driver ()`
- `bool set_stereo_output_configuration (const stereo_output_config_t& config)`
- `int get_stereo_output_configuration ()`
- `bool use_analog_microphone (const bool& left = true, const bool& right = true)`
- `bool use_digital_microphone (const bool &left = true, const bool& right = true)`
- `int is_left_microphone_analog ()`
- `int is_right_microphone_analog ()`
- `bool enable_i2c_error_detection (const bool& en = true)`
- `bool disable_i2c_error_detection ()`
- `int is_i2c_error_detection_enabled ()`
- `int i2c_error_occured ()`
- `bool set_audio_interface_mode (const audio_interface_mode_t& mode)`
- `int get_audio_interface_mode ()`
- `bool set_audio_interface_word_length (const audio_interface_word_length_t& mode)`
- `int get_audio_interface_word_length ()`
- `bool software_reset ()`
- `void shutdown ()`
- `void print_settings ()`
- `int read_ADC_flags ()`
- `i2s_t& data_interface ()`
- `bool empty () const`
Viz A.3 I2S
- `uint32_t read () const`
Viz A.3 I2S
- `void write (const uint32_t& data)`
Viz A.3 I2S

Kromě těchto metod jsou definovány následující objekty:

- `clock`
Nastavení hodin kodeku.
 - `enum clk_in_source_t`
 - `MCLK = 0`
 - `GPI02 = 1`
 - `BCLK = 2`
 - `enum clk_in_selectin_t`
 - `PLLDIV_OUT = 0`
 - `CLKDIV_OUT = 1`
 - `bool set_clk_div_source(const clk_in_source_t& source)`
 - `int get_clk_div_source()`
 - `bool set_clk_pll_source(const clk_in_source_t& source)`
 - `int get_clk_pll_source()`
 - `bool set_codec_clock_source(const clk_in_selectin_t& source)`
 - `int get_codec_clock_source()`
 - `bool enable_pll(const bool& en = true)`
 - `bool disable_pll()`
 - `int is_pll_enabled()`
 - `bool set_div_Q(const uint8_t& value)`

- int get_div_Q()
- bool set_pll_P(const uint8_t& value)
- int get_pll_P()
- bool set_pll_R(uint8_t value)
- int get_pll_R()
- bool set_pll_J(const uint8_t& value)
- bool set_pll_D(const uint16_t& value)
- int get_pll_D()
- bool set_pll(const uint8_t& J, const uint16_t& D, const uint8_t& R, const uint8_t& P, const bool& enable = true)
- bool set_bclk_slave_mode()
- bool set_bclk_master_mode()
- int is_bclk_master()
- bool set_wclk_slave_mode()
- bool set_wclk_master_mode()
- int is_wclk_master()
- bool enable_3_state_dout(const bool& en = true)
- bool disable_3_state_dout()
- int is_3_state_dout_enabled()
- bool enable_continuous_clock_output(const bool& en = true)
- bool disable_continuous_clock_output()
- int is_continuous_clock_output_enabled()
- bool enable_256_clock_transfer_mode(const bool& en = true)
- bool disable_256_clock_transfer_mode()
- int is_256_clock_transfer_mode_enabled()
- bool set_data_offset(const uint8_t& value)
- int get_data_offset()
- left_adc
- right_adc

Nastavení AD převodníků.

- enum high_pass_filter_settings_t
 - DISABLED = 0
 - HPF0045FS = 1
 - HPF0125FS = 2
 - HPF025FS = 3

Nastavení filtru typu horní propust. Možnosti jsou: vypnuto, $0,0045 \times f_{S(\text{ref})}$, $0,0125 \times f_{S(\text{ref})}$ a $0,025 \times f_{S(\text{ref})}$.

- volume_t – viz A.5 Ovládání hlasitosti
- bool power_on(const bool& on = true)
- bool power_off()
- int is_powered()
- int was_overflow()
- bool enable_resync(const bool& en = true)
- bool disable_resync()
- int is_resync_enabled()
- bool use_high_pass_filter_default_coeffs()
- bool use_high_pass_filter_programmable_coeffs()
- int using_high_pass_filter_default_coeffs()
- bool set_high_pass_filter(const high_pass_filter_settings_t& settings)
- int get_high_pass_filter_settings()
- bool use_programmable_filters(const bool& use = true)
- int using_programmable_filters()

- line2_L_to_left_adc
line2_R_to_left_adc
line2_L_to_right_adc
line2_R_to_right_adc

Nastavení propojení mezi vstupy a AD převodníky.

- bool set_volume(const uint8_t& value)

Hodnoty v rozsahu 0 ÷ 8. Výsledné zesílení:

$$A = -1,5 \times value [dB] \quad (1)$$

Je-li zadána větší hodnota, je vstup odpojen.

- int get_volume()
- bool mute(const bool& mute = true)
- bool unmute()
- int is_muted()
- line1_LP_to_left_adc
line1_RP_to_left_adc
line1_RP_to_right_adc
line1_LP_to_right_adc

Nastavení propojení mezi diferenciálními vstupy a AD převodníky. Disponují stejným nastavením hlasitosti, jako ostatní vstupy (viz předchozí bod). Navíc obsahují následující metody:

- bool set_single_ended()
- bool set_differential()
- int is_differential()

- left_agc
right_agc

Nastavení automatické kontroly hlasitosti.

- enum target_level_t
 - A5_5dB = 0
 - A8dB = 1
 - A10dB = 2
 - A12dB = 3
 - A14dB = 4
 - A17dB = 5
 - A20dB = 6
 - A24dB = 7
- enum attack_time_t
 - At8ms = 0
 - At11ms = 1
 - At16ms = 2
 - At20ms = 3
- enum new_attack_time_t
 - NAT7ms = 0
 - NAT8ms = 1
 - NAT10ms = 2
 - NAT11ms = 3
- enum decay_time_t
 - Dt100ms = 0
 - Dt200ms = 1
 - Dt400ms = 2

- Dt500ms = 3
- o enum new_decay_time_t
 - NDt50ms = 0
 - NDt150ms = 1
 - NDt250ms = 2
 - NDt350ms = 3
- o enum noise_gate_hysteresis_t
 - H1dB = 0
 - H2dB = 1
 - H3dB = 2
 - DISABLED = 3
- o bool enable(const bool& en = true)
- o bool disable()
- o int is_enabled()
- o bool set_target_level(const target_level_t& level)
- o int get_target_level()
- o bool set_attack_time(const attack_time_t& time)
- o int get_attack_time()
- o bool set_decay_time(const decay_time_t& time)
- o int get_decay_time()
- o bool set_max_gain(const uint8_t& gain)
- o int get_max_gain()
- o bool set_noise_gate_hysteresis(const noise_gate_hysteresis_t& hysteresis)
- o int get_noise_gate_hysteresis()
- o bool set_noise_gate_threshold(const uint8_t& threshold)
- o int get_noise_gate_threshold()
- o bool enable_clip_stepping(const bool& en = true)
- o bool disable_clip_stepping()
- o int is_clip_stepping_enabled()
- o int get_applied_gain()
- o bool set_noise_detection_debounce(const uint8_t& debounce)
- o int get_noise_detection_debounce()
- o bool set_signal_detection_debounce(const uint8_t& debounce)
- o int get_signal_detection_debounce()
- o bool use_new_attack_time()
- o bool use_attack_time()
- o int using_new_attack_time()
- o bool set_new_attack_time(const new_attack_time_t& time)
- o int get_new_attack_time()
- o bool set_new_attack_time_multiplication(const uint8_t& time)
- o int get_new_attack_time_multiplication()
- o bool use_new_decay_time()
- o bool use_decay_time()
- o int using_new_decay_time()
- o bool set_new_decay_time(const new_decay_time_t& time)
- o int get_new_decay_time()
- o bool set_new_decay_time_multiplication(const uint8_t& time)
- o int get_new_decay_time_multiplication()

- left_dac
right_dac

Nastavení DA převodníků

- o enum channel_selection_t
 - MUTE = 0
 - THIS_CHANNEL = 1
 - SECOND_CHANNEL = 2
 - MONO_MIX = 3
 - o enum output_path_t
 - MIX_INPUT = 0
 - LINE_INPUT = 1
 - HIGH_POWER_INPUT = 2
 - o enum master_volume_t
 - INDEPENDENT = 0
 - RIGHT = 1
 - LEFT = 2
 - o enum quescent_current_t
 - DEFAULT = 0
 - PLUS_50 = 1
 - PLUS_100 = 3
 - o volume_t – viz A.5 Ovládání hlasitosti
 - o bool set_channel(const channel_selection_t& channel)
 - o int get_channel()
 - o bool power_on(const bool& on = true)
 - o bool power_off()
 - o int is_powered()
 - o int was_overflow()
 - o bool set_output_path(const output_path_t& path)
 - o int get_output_path()
 - o bool set_master_volume(const master_volume_t& master)
 - o int get_master_volume()
 - o bool set_quescent_current(const quescent_current_t& increase)
 - o int get_quescent_current()
 - o bool enable_digital_effects_filter(const bool& en = true)
 - o bool disable_digital_effects_filter()
 - o int is_digital_effects_filter_enabled()
 - o bool enable_deemphasis_filter(const bool& en = true)
 - o bool disable_deemphasis_filter()
 - o int is_deemphasis_filter_enabled()
 - o bool enable_resync(const bool& en = true)
 - o bool disable_resync()
 - o int is_resync_enabled()
- line_out_L
line_out_R

Nastavení signálových výstupů.

- o bool set_volume(const uint8_t& value)
 - value – hodnota v rozsahu 0 ÷ 9 dB.
 - o int get_volume()
 - o bool mute(const bool& mute = true)
 - o bool unmute()
 - o int is_muted()
 - o int are_gains_applied()
 - o bool power_on(const bool& value = true)
 - o bool power_off()
 - o int is_powered()

- Následující objekty jsou typu `volume_t` (viz A.5 Ovládání hlasitosti). Ovládají amplitudu přivedeného z daného bloku na výstup.
 - `pga_L`
 - `dac_L1`
 - `pga_R`
 - `dac_R1`

- `hplout`
`hprout`

Nastavení silových výstupů. Obsahují stejná nastavení jako signálové výstupy.

Navíc obsahují následující datové typy a metody:

- enum `common_mode_voltage_t`
 - `CMV_1V35` = 0
 - `CMV_1V5` = 1
 - `CMV_1V65` = 2
 - `CMV_1V8` = 3
- enum `soft_stepping_t`
 - `ONE_FS` = 0
 - `TWO_FS` = 1
 - `DISABLED` = 2
- bool `set_high_impedance()`
- bool `set_weak_drive()`
- int `is_high_impedance()`
- bool `enable_short_circuit_protection(const bool& en = true)`
- bool `disable_short_circuit_protection()`
- int `is_short_circuit_protection_enabled()`
- bool `set_short_circuit_shutdown()`
- bool `set_short_circuit_current_limit()`
- int `is_short_circuit_shutdown()`
- int `is_short_circuit_detected()`
- int `was_short_circuit_detected()`
- bool `set_common_mode_voltage(const common_mode_voltage_t& voltage)`
- int `get_common_mode_voltage()`
- bool `set_volume_soft_stepping(const soft_stepping_t& soft_stepping)`
- int `get_volume_soft_stepping()`
- bool `set_AVDD_as_weak_driver()`
- bool `set_bandgap_as_weak_driver()`
- int `is_AVDD_weak_driver()`
- bool `set_power_on_delay(const uint8_t& delay)`
- int `get_power_on_delay()`
- bool `set_driver_ramp_up_step_time(const uint8_t& value)`
- int `get_driver_ramp_up_step_time()`

- `hplcom`

Nastavení levého komplementárního silového výstupu. Obsahuje stejná nastavení jako silové výstupy a navíc následující:

- enum `mode_t`
 - `DIFF_HPLOUT` = 0
 - `VCM` = 1
 - `SINGLE_ENDED` = 2

- o `bool set_mode(const mode_t& mode)`
 - o `int get_mode()`
 - **hprcom**
 Nastavení pravého komplementárního silového výstupu. Obsahuje stejná nastavení jako silové výstupy a navíc následující:
 - o `enum mode_t`
 - `DIFF_HPROUT` = 0
 - `VCM` = 1
 - `SINGLE_ENDED` = 2
 - `DIFF_HPLCOM` = 3
 - `HPLCOM_FEEDBACK` = 4
 - o `bool set_mode(const mode_t& mode)`
 - o `int get_mode()`
 - **line2_RP**
line1_RM
line1_RP
line2_LP
line1_LM
line1_LP
 Nastavení přímého analogového propojení když je kodek v úsporném režimu.
 - o `bool enable(const bool& en = true)`
 - o `bool disable()`
 - o `int is_enabled()`
 - **filters**
 Sdružuje nastavení koeficientů digitálních filtrů. Každý filtr se skládá z koeficientů čitatele N a koeficientů jmenovatele D. Koeficienty jsou přístupné přes metody filtrů:
 - o `coeff_t& N(const uint8_t& index)`
 - o `coeff_t& D(const uint8_t& index)`
 Vrácený objekt typu `coeff_t` má následující metody, pomocí kterých se dá s hodnotou daného koeficientu manipulovat:
 - o `bool set(const int16_t& value)`
 - o `int get()`
 Jsou definovány následující filtry:
 - o `left_effects`
`right_effects`
 Koeficienty N0, N1, N2, N3, N4, N5, D1, D2, D4, D5
 - o `left_de_emphasis`
`right_de_emphasis`
 Koeficienty N0, N1, D1
 - o `left_high_pass`
`right_high_pass`
 Koeficienty N0, N1, D1
- Příklad použití:**
- ```
audio_codec.filters.left_effects.N(0).set(1234);
```
- Zápis do neexistujícího koeficientu nemá žádný efekt a je vždy úspěšný (návratová hodnota je `true`). Čtení neexistujícího koeficientu vrátí hodnotu 0. Selže-li funkce pro čtení, je vrácena hodnota 32768.

- `attenuation_3D`  
objekt typu `coeff_t`, na kterém jsou definovány následující metody:
  - `bool set(const int16_t& value)`
  - `int get()`  
Selže-li funkce pro čtení, je vrácena hodnota 32768.

## A.5 Ovládání hlasitosti

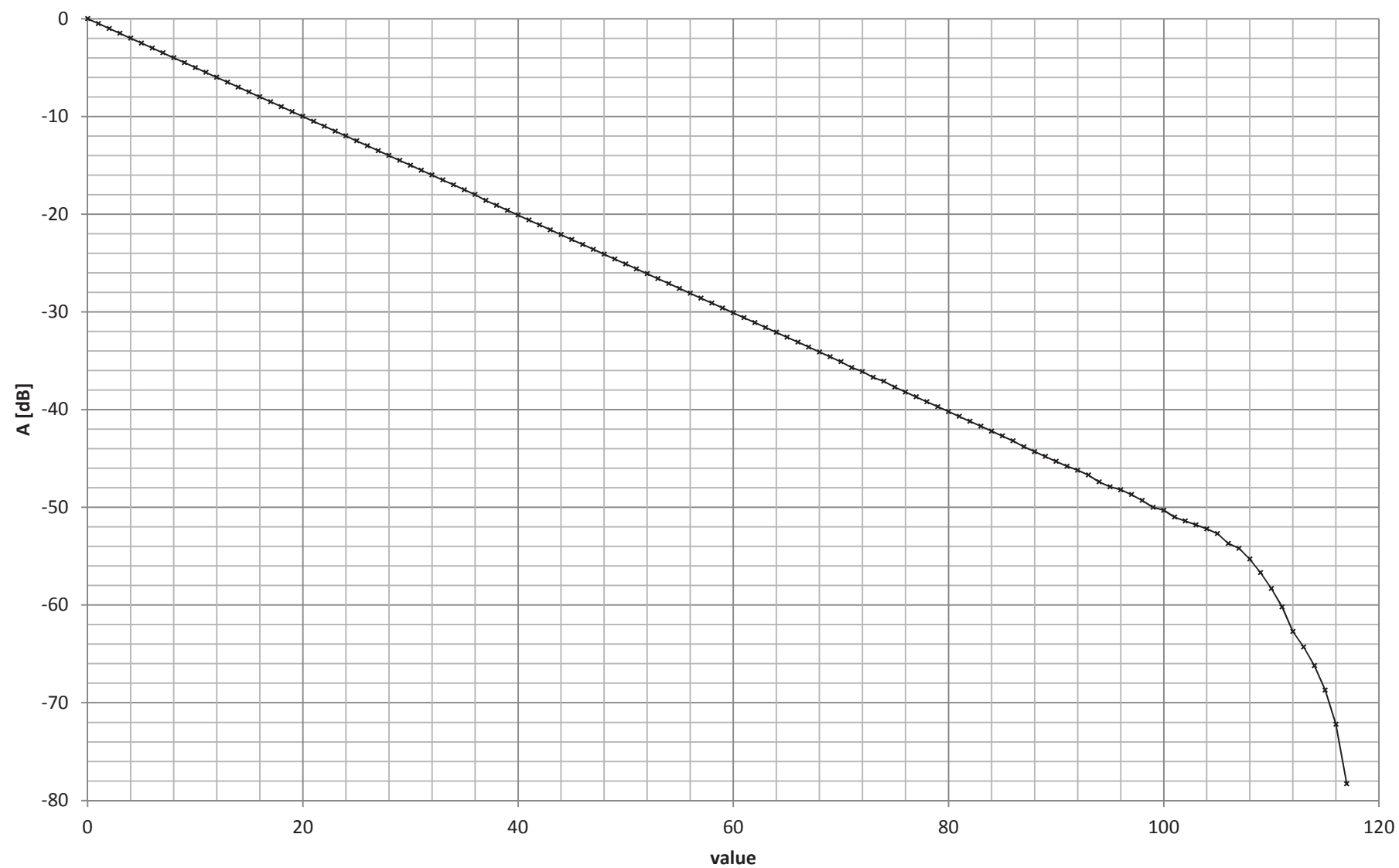
Všechny objekty typu `volume_t`, sloužící k ovládání hlasitosti, mají následující metody:

- `bool set_volume(const uint8_t& value)` - viz následující tabulka
- `int get_volume()`
- `bool mute(const bool& mute = true)`
- `bool unmute()`
- `int is_muted()`

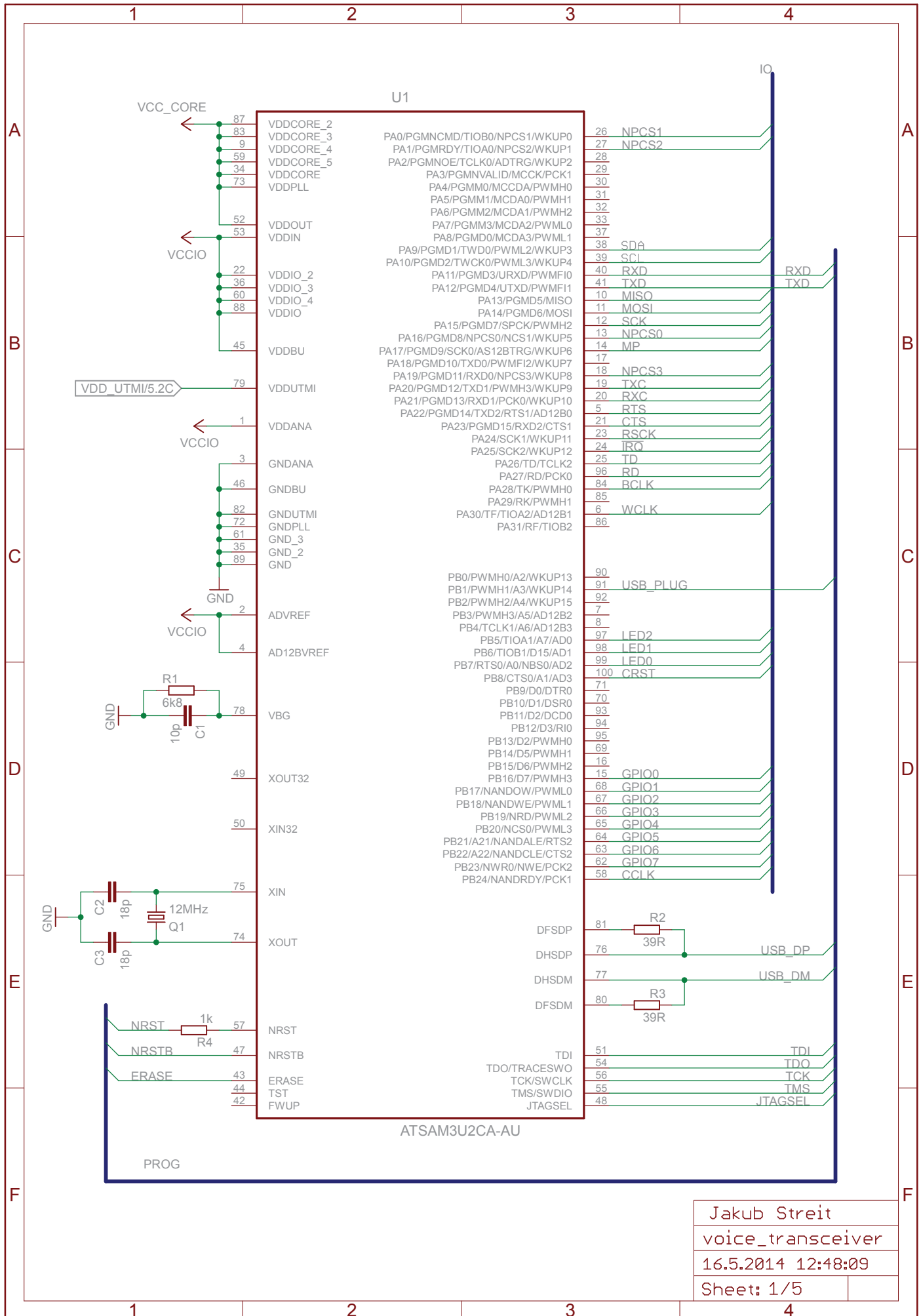
| value | Gain<br>[dB] | value | Gain<br>[dB] | value | Gain<br>[dB] | value        | Gain<br>[dB] |
|-------|--------------|-------|--------------|-------|--------------|--------------|--------------|
| 0     | 0,0          | 30    | -15,0        | 60    | -30,1        | 90           | -45,3        |
| 1     | -0,5         | 31    | -15,5        | 61    | -30,6        | 91           | -45,8        |
| 2     | -1,0         | 32    | -16,0        | 62    | -31,1        | 92           | -46,2        |
| 3     | -1,5         | 33    | -16,5        | 63    | -31,6        | 93           | -46,7        |
| 4     | -2,0         | 34    | -17,0        | 64    | -32,1        | 94           | -47,4        |
| 5     | -2,5         | 35    | -17,5        | 65    | -32,6        | 95           | -47,9        |
| 6     | -3,0         | 36    | -18,0        | 66    | -33,1        | 96           | -48,2        |
| 7     | -3,5         | 37    | -18,6        | 67    | -33,6        | 97           | -48,7        |
| 8     | -4,0         | 38    | -19,1        | 68    | -34,1        | 98           | -49,3        |
| 9     | -4,5         | 39    | -19,6        | 69    | -34,6        | 99           | -50,0        |
| 10    | -5,0         | 40    | -20,1        | 70    | -35,1        | 100          | -50,3        |
| 11    | -5,5         | 41    | -20,6        | 71    | -35,7        | 101          | -51,0        |
| 12    | -6,0         | 42    | -21,1        | 72    | -36,1        | 102          | -51,4        |
| 13    | -6,5         | 43    | -21,6        | 73    | -36,7        | 103          | -51,8        |
| 14    | -7,0         | 44    | -22,1        | 74    | -37,1        | 104          | -52,2        |
| 15    | -7,5         | 45    | -22,6        | 75    | -37,7        | 105          | -52,7        |
| 16    | -8,0         | 46    | -23,1        | 76    | -38,2        | 106          | -53,7        |
| 17    | -8,5         | 47    | -23,6        | 77    | -38,7        | 107          | -54,2        |
| 18    | -9,0         | 48    | -24,1        | 78    | -39,2        | 108          | -55,3        |
| 19    | -9,5         | 49    | -24,6        | 79    | -39,7        | 109          | -56,7        |
| 20    | -10,0        | 50    | -25,1        | 80    | -40,2        | 110          | -58,3        |
| 21    | -10,5        | 51    | -25,6        | 81    | -40,7        | 111          | -60,2        |
| 22    | -11,0        | 52    | -26,1        | 82    | -41,2        | 112          | -62,7        |
| 23    | -11,5        | 53    | -26,6        | 83    | -41,7        | 113          | -64,3        |
| 24    | -12,0        | 54    | -27,1        | 84    | -42,2        | 114          | -66,2        |
| 25    | -12,5        | 55    | -27,6        | 85    | -42,7        | 115          | -68,7        |
| 26    | -13,0        | 56    | -28,1        | 86    | -43,2        | 116          | -72,2        |
| 27    | -13,5        | 57    | -28,6        | 87    | -43,8        | 117          | -78,3        |
| 28    | -14,0        | 58    | -29,1        | 88    | -44,3        | 118 -<br>127 | Mute         |
| 29    | -14,5        | 59    | -29,6        | 89    | -44,8        |              |              |

Tabulka závislost skutečného zesílení na nastavené hodnotě

### Závislost skutečného zesílení na nastavené hodnotě



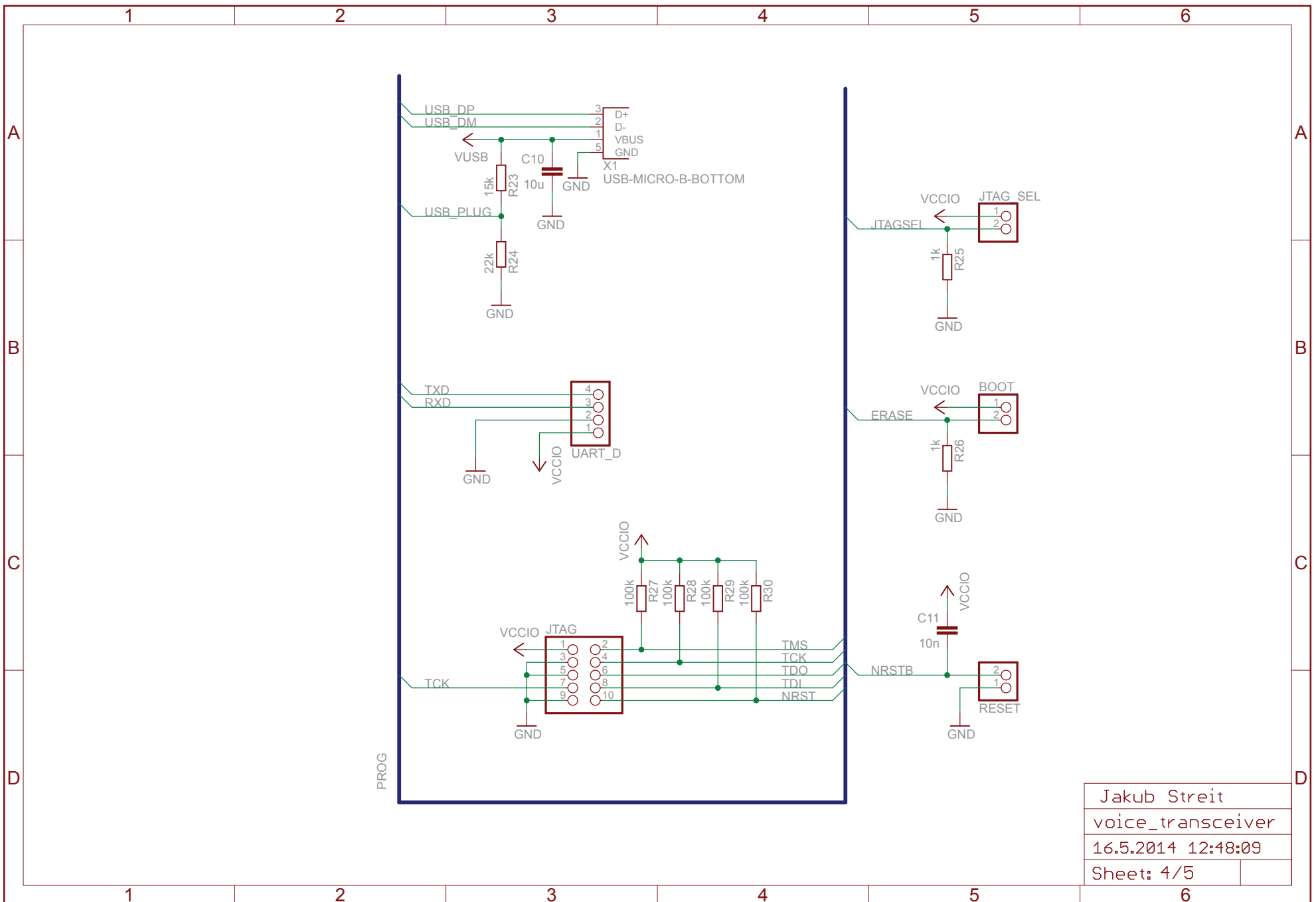
## **B SCHÉMA ZAPOJENÍ**











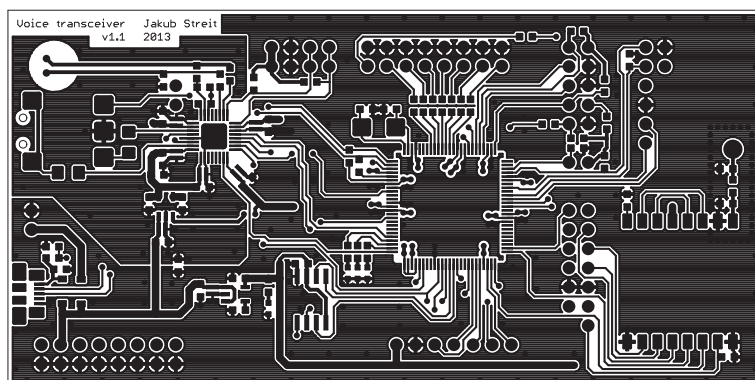


## C SEZNAM SOUČÁSTEK

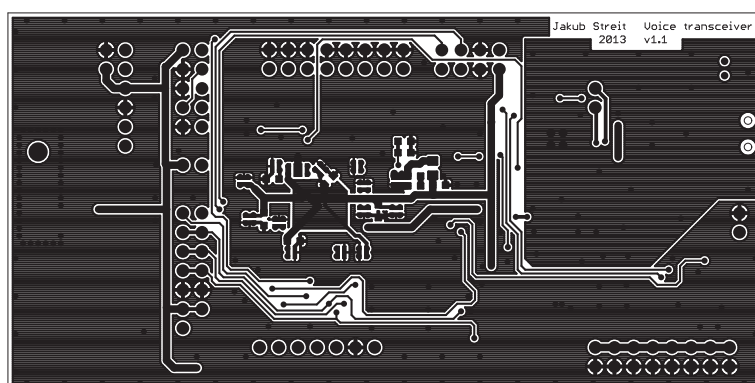
| Rezistory          |       |              |          |                                                                                      |                                                                                                              |
|--------------------|-------|--------------|----------|--------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| Hodnota            |       | Pouzdro      | Počet    | Označení součástek                                                                   |                                                                                                              |
| 0R                 |       | SMD 0603     | 2        | R33, RCL1                                                                            |                                                                                                              |
| 1R                 |       | SMD 0603     | 1        | R32                                                                                  |                                                                                                              |
| 39R                |       | SMD 0603     | 2        | R2, R3                                                                               |                                                                                                              |
| 1k                 |       | SMD 0603     | 18       | R4, R5, R8, R9, R12, R13, R14, R15, R16, R17, R18, R19, R20, R21, R22, R25, R26, R31 |                                                                                                              |
| 4k7                |       | SMD 0603     | 2        | R6, R7                                                                               |                                                                                                              |
| 6k8                |       | SMD 0603     | 1        | R1                                                                                   |                                                                                                              |
| 10k                |       | SMD 0603     | 1        | R10, R11                                                                             |                                                                                                              |
| 15k                |       | SMD 0603     | 1        | R23                                                                                  |                                                                                                              |
| 22k                |       | SMD 0603     | 1        | R24                                                                                  |                                                                                                              |
| 100k               |       | SMD 0603     | 4        | R27, R28, R29, R30                                                                   |                                                                                                              |
| Kondenzátory       |       |              |          |                                                                                      |                                                                                                              |
| C [F]              | U [V] | Dielektrikum | Pouzdro  | Počet                                                                                | Označení součástek                                                                                           |
| 10p                | 50    | C0G (NP0)    | SMD 0603 | 1                                                                                    | C1                                                                                                           |
| 18p                | 50    | C0G (NP0)    | SMD 0603 | 2                                                                                    | C2, C3                                                                                                       |
| 10n                | 50    | X7R          | SMD 0603 | 3                                                                                    | C11, C25, C51                                                                                                |
| 100n               | 25    | X7R          | SMD 0603 | 4                                                                                    | C4, C5, C6, C9                                                                                               |
| 100n               | 10    | X7R          | SMD 0306 | 22                                                                                   | C12, C13, C14, C15, C17, C19, C21, C24, C28, C29, C30, C31, C32, C33, C34, C35, C39, C40, C42, C45, C48, C50 |
| 1u                 | 25    | X5R          | SMD 0603 | 5                                                                                    | C38, C41, C43, C46, C47                                                                                      |
| 2u2                | 16    | X5R          | SMD 0603 | 4                                                                                    | C23, C26, C49, C52                                                                                           |
| 4u7                | 16    | X5R          | SMD 0603 | 6                                                                                    | C16, C18, C20, C22, C36, C37                                                                                 |
| 10u                | 10    | X5R          | SMD 0603 | 3                                                                                    | C10, C27, C44                                                                                                |
| 100u               | 6,3   | Tantal       | SMD 1206 | 2                                                                                    | C7, C8                                                                                                       |
| Diody              |       |              |          |                                                                                      |                                                                                                              |
| Typ                |       | Pouzdro      | Počet    | Označení součástek                                                                   |                                                                                                              |
| LED zelená         |       | SMD 0603     | 2        | LED1, PWRLED                                                                         |                                                                                                              |
| LED žlutá          |       | SMD 0603     | 1        | LED2                                                                                 |                                                                                                              |
| LED červená        |       | SMD 0603     | 1        | LED3                                                                                 |                                                                                                              |
| Shottky PMEG2010ER |       | SOD123FL     | 2        | D1, D2                                                                               |                                                                                                              |

| Konektory              |                |                              |       |                    |
|------------------------|----------------|------------------------------|-------|--------------------|
| Typ                    | Počet          | Označení součástek           |       |                    |
| Jack 3,5 mm F 4pin DPS | 1              | AUDIO                        |       |                    |
| USB-MICRO-B-BOTTOM     | 1              | X1                           |       |                    |
| PINHD-1X1              | 1              | RFSHDN                       |       |                    |
| PINHD-1X2              | 3              | BOOT, JTAG_SEL, RESET        |       |                    |
| PINHD-1X2/90           | 2              | REPRO, BAT                   |       |                    |
| PINHD-1X4              | 1              | UART_D                       |       |                    |
| PINHD-1X7              | 1              | USART                        |       |                    |
| PINHD-1X7              | 1              | I2C                          |       |                    |
| PINHD-2X5              | 1              | JTAG                         |       |                    |
| PINHD-2X6              | 1              | SPI                          |       |                    |
| PINHD-2X8              | 2              | GPIO, PWR                    |       |                    |
| Ostatní součástky      |                |                              |       |                    |
| Typ                    | Označení       | Pouzdro                      | Počet | Označení součástek |
| cívka                  | 10 uH 200 mA   | SMD 0805                     | 1     | L1                 |
| krystal                | 12 MHz         | 5×3,2 mm                     | 1     | Q1                 |
| mikrofon               | WM-62-PC       | d6mm                         | 1     | MIC                |
| stabilizátor           | TPS73033DBV    | SOT23-DBV                    | 2     | IC1, IC2           |
| paměť FRAM             | FM25CL64B-G    | SO08                         | 1     | U3                 |
| Audio kodek            | TLV320AIC3101  | QFN32                        | 1     | U2                 |
| RF modul               | RFM22-868-S1/2 | RFM22-XXX-S2                 | 1     | RF                 |
| mikroprocesor          | ATSAM3U2CA-AU  | QFP50P1600X1600X<br>160-100N | 1     | U1                 |

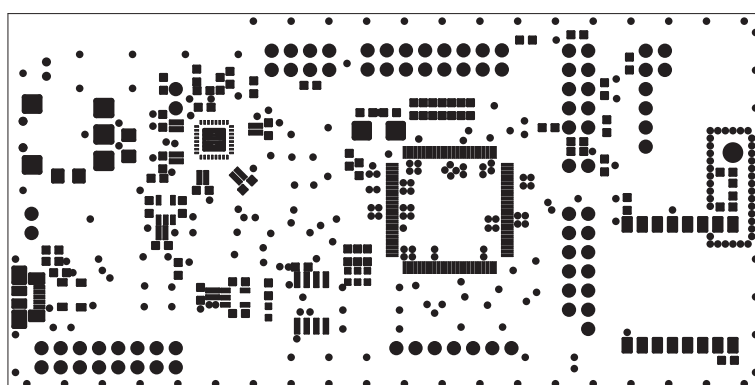
## D DPS



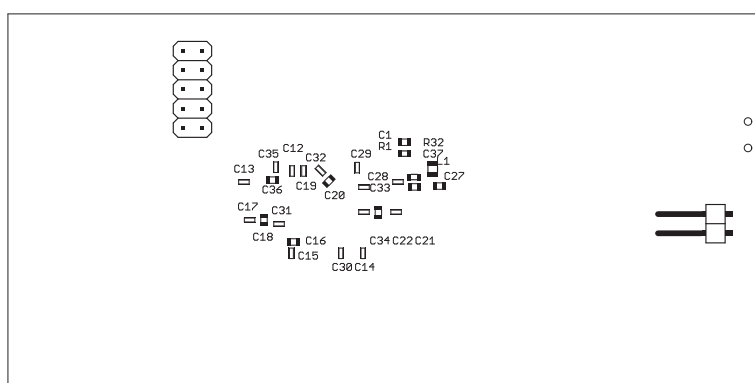
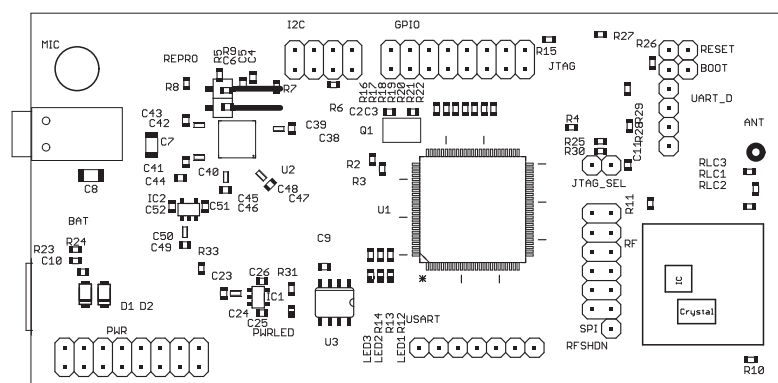
Strana součástek (top)



Strana spojů (bottom)



Strana součástek (top) – nepájivá maska





## **E OBSAH CD**

- Elektronická verze tohoto textu ve formátu pdf a docx
- Návrh DPS v programu Eagle 6.4 (soubory sch a brd)
- Zdrojové texty programu pro mikroprocesor v jazyce C/C++, včetně projektového souboru Atmel Studio 6.2 a nezbytných skriptů v jazyce Python 2.7
- Zdrojové texty programu Lorris (C/C++, Qt), včetně projektového souboru QtCreatoru
- Program Lorris (MS Windows)