

Заголовочная часть АСТ файлов

Содержание:

1. Заголовок файла АСТ1-формата	1
2. Файл заголовка АСТConv.h	4

1. Заголовок файла АСТ1-формата

№ гр.	№ эл-та	Смещение	Длина	Тип	Образец, комментарии
1	0¹	0	7		[АСТ1СТ_- Формат заголовка (1.0)]
	1	0	4		АСТ1 – Идентификатор формата (протокола) заголовка
	2	4	2		Модальность. СТ – компьютерные томограммы
	3	6	1		разделитель (пробел)
2	0	7	15		[a0011c12.015F*_] – Идентификатор файла
	1	7	1		a – индекс БД (значение зависит от организации)
	2	8	4		0011 – рег. Номер пациента в БД
	3	12	1		c – код типа данных (c – КТ, d – дозы и т.п.)
	4	13	1		1 – номер исследования (от 0 до 9)
	5	14	1		2 – номер серии в исследовании (от 0 до 9)
	6	15	1		'.' – разделитель (точка)
	7	16	3		015 – номер изображения в серии
	8	19	2		F* – резерв
	9	21	1		разделитель (пробел)
3	0	22	18		[3434>0512x0512bro_] – Формат записи данных
	1	22	4		3434 – смещение данных от начала файла
	2	26	1		> – разделитель
	3	27	4		0512 – число строк в изображении
	4	31	1		x – разделитель
	5	32	4		0512 – число столбцов в изображении
	6	36	1		b – число байт, отведенных для записи пиксела в данных. Принимает фиксированные значения: W – для 2 байт/пиксел, B – для 1 байта на пиксел
	7	37	1		r – определяет порядок байтов (только для 16 и 32 разрядных чисел) и представление целых (со знаком или без) ² :

¹ Элемент группы с номером 0 означает "вся группа"

№ гр.	№ эл-та	Смещение	Длина	Тип	Образец, комментарии
					$r = 0$ – целое без знака, порядок байтов не инвертирован, 12 значащих разрядов; $r = 1$ – целое без знака, порядок байтов инвертирован; $r = 2$ – целое со знаком, порядок байтов не инвертирован; $r = 3$ – целое со знаком, порядок байтов инвертирован.
	8	38	1		o – маска оверлея. Имеет смысл только для W0* . Шестнадцатиричное число (0 – 15), задающее маску наличия оверлеев в младших (старших?) разрядах данных. Например, $o = 0x0A_{16} = 10_{10} = 1010B_2$ означает, что в 15 и 13 разрядах данных содержится информация о ROI (областях интереса, в т.ч. мишенях) ³ .
	9	39	1		разделитель (пробел)
4	0	40	25		[d-1024u+3071b-1500c-0950_] – числовой диапазона данных
	1	40	6		d-1024 – минимальное значение (не включает "заполнитель").
	2	46	6		u+3071 – максимальное значение.
	3	52	6		b-1500 – значение "заполнителя" (служит для дополнения круглой области восстановленных данных до прямоугольника).
	4	58	6		c-0950 – пороговое значение (код для "чистки" поля зрения вне зоны объекта) ⁴ .
	5	64	1		разделитель (пробел)
5	0	65	15		[S1a-0997w+0003_] – Определитель шкалы СТ данных
	1	65	2		S^* – тип шкалы: $S0$ – 12-разрядная СТ-шкала рентгеновских плотностей (воздух, вода) = (0, 1000); $S1$ – стандартная шкала Хаунсфилда ⁵ (-1000, 0); $S2$ – шкала $S3$ – шкала

² Порядок байтов во входном потоке считается инвертированным, если первым поступают наименее значимые байты (как для Intel процессора). Это так называемый Little Endian упорядочивание в отличие от Big Endian (напр. процессор Macintosh). Этот порядок имеет значение только для 16 и 32 разрядных чисел. Для Intel не требуется каких-то дополнительных преобразований для чтения/записи чисел в формате Little Endian.

³ Смысл, придаваемый каждой из оверлейных плоскостей, выходит за рамки настоящего документа и определяется соглашениями системы Прогам/Проком.

⁴ Код c-9999 информирует потребителя АСТ файла об отсутствии рекомендаций по "очистке" (замены не производятся). Коды < -0024 фактически соответствуют "чистке" воздуха от выбросов. Естественно, бланк заменять при "чистке" нецелесообразно!

⁵ Используется формат W2/W3 для записи данных.

№ гр.	№ эл-та	Смещение	Длина	Тип	Образец, комментарии
	2	67	12		Значение поля зависит от формата: a-0997w+0003 – для шкал S0, S1, S2 калибровочные значения для воздуха и воды; i*****.*** – для шкалы S3 имя индексной таблицы (LUT).
	3	79	1		разделитель (пробел)
6	0	80	12		[Z+0600F2500_] – Укладка пациента и пространственная привязка слоя
	1	80	1		Z – Символ, определяющий продольное положение пациента: Z = H – головой вперед (Head First) Z = F – ногами вперед (Feet First)
	2	81	5		+0600 – смещение текущего слоя относительно базы в дмм
	3	86	1		F – символ, определяющий положение пациента на деке томограф. может принимать значения: F = S (или F) – на спине (Supine) F = P – на животе, ничком (Prone) F = L – на левом боку (Left) F = R – на правом боку (Right)
	4	87	4		2500 – поле зрения, дмм
	4	91	1		разделитель (пробел)
7	0	92	11		[40s030i030_] – параметры визуализации
	1	92	2		40 – число слоев в 16 ричной форме (до 255 слоев)
	2	94	4		s030 – толщина слоя, дмм
	3	98	4		i030 – расстояние между центрами слоев ⁶ , дмм
	4	102	1		разделитель (пробел)
8	0	103	3		[**_] – угол наклона гантри
	1	103	2		** -- угол наклона гантри
	2	105	1		разделитель (пробел)
9	0	106	12		– параметры визуализации
	1	106	6		L+0085 – уровень
	2	112	5		W0200 – окно
	3	117	1		разделитель (пробел)
10	0	118	10		[2002TOR00→] – конец заголовка.
	1	118	9		2002TOR00 – авторизация
	2	127	1		код 26 ₁₀ – для варианта записи с сохранением заголовка исходного файла

⁶ Для последнего слоя – расстояние до предыдущего слоя, для остальных – расстояние до следующего слоя

2. Файл заголовка ACTConv.h

```
#ifndef ACTCONVH
#define ACTCONVH
```

```
/*
```

ACTConv.h содержит макро и прототипы функций, содержащихся в библиотеке (динамической)преобразования томографического файла во внутренний формат пакета PROCOM (АСТ - формат).

Каждая библиотека должна содержать эти функции. Описания функций - в CTRd.h

При построении библиотеки следует определить __BUILDING_THE_DLL

При использовании DLL, загружаемых во время выполнения программы перед строкой

```
#include "CTRd.h"
```

необходимо включить строку

```
#define RUN_TIME_LOADED_DLL.
```

В этом случае заголовок определяет тип указателя на функцию: PFN<имя ф-ции>.

Типы WINAPI и extern "C" обеспечивают передачу в dll имени функции без изменений (в т.ч. без добавления символа подчеркивания).

WINAPI эквивалентно FAR PASCAL.

Передача строк (кроме имени и описания б-ки) с помощью функций, имеющих тип BOOL и одну или несколько пар аргументов - адрес длины строки и адрес переменной с адресом строки. Строка не обязательно должна заканчиваться нулем, адрес ее теряет смысл после загрузки другого томографического файла или выгрузки библиотеки, т.е. строку следует скопировать после ее получения.

Это сделано для упрощения ввода (не надо запрашивать длину строки или задавать максимальную длину строки).

```
*/
```

```
#ifdef __BUILDING_THE_DLL
    #define __EXPORT_TYPE(type, fnName) __declspec( dllexport ) type
    fnName
#else
```

```
#ifdef RUN_TIME_LOADED_ACT_DLL
#define __EXPORT_TYPE(type, fnName) typedef type (*
PF_##fnName )
#else
#define __EXPORT_TYPE(type, fnName) __declspec( dllimport )
type fnName
#endif
#endif
```

```
#include <stdlib.h>
```

```
// Определение констант для задания/тестирования ciFlag из структуры
ACTFILEINFOHEADER.
```

```
// параметр ciXXXX определен, если в ciFlag взведен бит e_ciXXXX:
```

```
#define e_ciPixelRepresentation      0x00000001
#define e_ciImageNumber              0x00000002
#define e_ciDataOff                  0x00000004
#define e_ciWidth                    0x00000008
#define e_ciHeight                   0x00000010
#define e_ciTransferSyntax            0x00000020
#define e_ciByteAlloc                0x00000040
#define e_ciBitsUsed                 0x00000080
#define e_ciOverlay                  0x00000100
#define e_ciMin                      0x00000200
#define e_ciMax                      0x00000400
#define e_ciPad                      0x00000800
#define e_ciCut                      0x00001000
#define e_ciTypeScale                0x00002000
#define e_ciAir                      0x00004000
#define e_ciWater                    0x00008000
#define e_ciZ                        0x00010000
#define e_ciF                        0x00020000
#define e_ciNoofSlices               0x00040000
#define e_ciZExt                     0x00080000
#define e_ciZStep                    0x00100000
#define e_ciLev                      0x00200000
#define e_ciWind                     0x00400000
#define e_ciOffACT                   0x00800000
#define e_ciImgOrient                0x01000000
#define e_ciIxTable                  0x02000000
#define e_ciPatientPosition          0x04000000
#define e_ciGantry                   0x08000000
#define e_ciSeriesNumber             0x10000000
```

```
#define MAX_CT_LIBNAME 16
```

// Структура, содержащая информацию, необходимую для преобразования в АСТ формат.

// Шестнадцатеричное число в комментарии соответствует тэгу в описании DICOM.

```
typedef struct {
    WORD ciSize;    // размер структуры
    int  ciLibNo;   // № библиотеки (тип ct файла): 0 – «сырой», 1 - DICOM, 2 -АСТ
    short ciPixelRepresentation; // 0/1 – unsigned/signed (0x00280103)
    DWORD ciFlag;   // если (ciFlag & e_ciXXX) == e_ciXXX, то
                    // параметр ciXXX определен
    short ciImageNumber; // № номер изображения (0x00200013)
    short ciSeriesNumber; // № серии (0x00200011)
    long ciDataOff;    // смещение данных в файле (0x7fe00010)
    WORD ciWidth;      // ширина изображения в пикселах (0x00280011)
    WORD ciHeight;     // ширина изображения в пикселах (0x00280010)
    BOOL ciTransferSyntax; // TRUE -- little-endian порядок (Intel 80x86 и DEC
    RISC)
    WORD ciByteAlloc;  // байты/пиксел: 2 or 1 (0x00280100)
    WORD ciBitsUsed;   // используемые биты : 16, 12 etc (0x00280101)
    BYTE ciOverlay;    // маска наличия оверлеев (в старших 4 разрядах)
    short ciMin;        // min ct величина
    short ciMax;        // max ct величина
    short ciPad;        // величина заполнителя
    short ciCut;        // уровень «обрезания»
    WORD ciTypeScale;   // 0 - ct, АСТ; 1 - Хаунсфилд
    short ciAir;        // Ct величина для воздуха (0x00281053; 0x00281052)
    short ciWater;      // Ct величина для воды (0x00281053; 0x00281052)
    short ciZ;          // положение изображения Z в 0.1mm (dmm):
    (0x0020003)
    short ciF;          // размер изображения в dmm (ширина, высота)
    short ciNoofSlices; // количество полученных изображений (0x00201002)
    short ciZExt;       // толщина среза, dmm (0x00180050)
    short ciZStep;      // расстояние до следующего среза
    short ciLev;        // центр «окна» (0x00281050)
    short ciWind;       // ширина «окна» (0x00281051)
    short ciOffACT;     // положение первого байта данных изображения в
    файле
    DWORD ciFileSize;
    union{
        struct{ // ориентация изображения (направляющие косинусы)
            float _ciImgRowCos[3]; // рядов
            float _ciImgColCos[3]; // колонок
        };
        float _ciImgRowColCos[6];
    };
};
```

```
    bool ciFlipLR;           // отображать зеркально (лево – право)
    bool ciFlipUD;           // отображать зеркально (верх – низ)
    char ciPatientPosition[2]; // положение пациента (как определено в АСТ)
    CHAR ciFileName[MAX_PATH]; // имя файла томограммы
    char ciIxTable[13];       // имя индексной табл. для ст-шкалы
    BOOL ciPacked;            // TRUE – упаковка по 12 бит.
    short      ciGantry;       // угол наклона гантри
} ACTFILEINFOHEADER, *LPACTFILEINFOHEADER;

//-----

#define e_ACTWO_ONLYDATA 0x0001 // Write only CT data
#define e_ACTWO_REPLACEFST 0x0002 // Replace 1st 128 byte

#ifdef __cplusplus
extern "C" {
#endif

    // имя типа преобразования (библиотеки):
    __EXPORT_TYPE(LPSTR, ACTCNV_GetLibName)(void);

    // описание преобразования (библиотеки):
    __EXPORT_TYPE(LPSTR, ACTCNV_GetLibDescription)(void);

    // TRUE, если формат файла соответствует преобразованию
    __EXPORT_TYPE(BOOL, ACTCNV_TestFile)(LPCTSTR lpszFileName);

    // Заполнение структуры ACTFILEINFOHEADER: [in] flnm, [out]
    ACTFILEINFOHEADER
    __EXPORT_TYPE(BOOL, ACTCNV_GetACTInfoHeader)(LPCTSTR flnm,
    LPACTFILEINFOHEADER pACTHdr);

#ifdef __cplusplus
}
#endif

// __cplusplus

#endif //ACTCONVH
```