| LCOV - code coverage report |
| --- |
|  |
| | Current view: | [top level](http://docs.google.com/index.html) - [PatternProcessor/include](http://docs.google.com/index.html) - PatternEdgeVectors.h (source / [functions](http://docs.google.com/PatternEdgeVectors.h.func.html)) |  |  | Hit | Total | Coverage | | --- | --- | --- | --- | --- | --- | --- | | Test: | EBM9K Converter Coverage Report |  | Lines: | 6 | 6 | 100.0 % | | Date: | 2012-03-23 |  | Functions: | 0 | 0 | - | | Legend: | Lines: hit not hit | Branches: + taken - not taken # not executed |  | Branches: | 0 | 0 | - | |  | |
|  |

|  |
| --- |
| Branch data Line data Source code  1 : : /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   2 : : \* Copyrighted 2006 by Interra Systems, USA  3 : : \* All rights reserved   4 : : \*  5 : : \* Date of Creation: Nov 2008  6 : : \* File Name: PatternEdgeVectors.h   7 : : \* Purpose: To specify the vectors along the non-parallel   8 : : \* edges of each pattern. Used for pattern slicing and division.  9 : : \* Author: Debarshi Kumar Sanyal  10 : : \*  11 : : \* Revision History:   12 : : \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/   13 : :   14 : : #ifndef \_PATTERNEDGEVECTORS\_H  15 : : #define \_PATTERNEDGEVECTORS\_H  16 : :   17 : : #include "PatternDivisionEnums.h"  18 : : #include "Position.h"  19 : :   20 : : namespace nft  21 : : {  22 : :   23 : : typedef struct \_EdgeVectorPair  24 : : {  25 : : //There are 2 vectors: E, K.  26 : : //Horizontal patterns: E vector along left edge  27 : : // K vector along right edge  28 : : //Vertical patterns: E vector along bottom edge  29 : : // K vector along top edge  30 : : IntPosition mE;  31 : : IntPosition mK;  32 : : } EdgeVectorPair;  33 : :   34 : : typedef struct PatternEdgeVectors  35 : : {  36 : : //There are 18 PDG-12X patterns  37 : : static const unsigned int MAX\_PATTERN\_CODES = 18;  38 : : static EdgeVectorPair mEdgeVectorRepo[MAX\_PATTERN\_CODES];  39 : :   40 : : public:  41 : :   42 : : static void initRepo();   43 : : //Initializes Pattern vector repository  44 : :   45 : : /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  46 : : \* Function Name: getEX   47 : : \* Purpose: Get the X component of the E vector of a given  48 : : \* pattern  49 : : \* Parameters:  50 : : \* code : uint32\_t [IN] [pattern code]  51 : : \* Return Values:   52 : : \* X component of the E vector of the pattern   53 : : \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  54 : : int32\_t getEX(uint32\_t code) const throw()  55 : 3753 : {  56 : : return PatternEdgeVectors::mEdgeVectorRepo[code].mE.getX();  57 : : }  58 : : /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  59 : : \* Function Name: getEY  60 : : \* Purpose: Get the Y component of the E vector of a given  61 : : \* pattern  62 : : \* Parameters:  63 : : \* code : uint32\_t [IN] [pattern code]  64 : : \* Return Values:  65 : : \* Y component of the E vector of the pattern  66 : : \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  67 : : int32\_t getEY(uint32\_t code) const throw()  68 : 7101 : {  69 : : return PatternEdgeVectors::mEdgeVectorRepo[code].mE.getY();  70 : : }  71 : : /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  72 : : \* Function Name: getKX  73 : : \* Purpose: Get the X component of the K vector of a given  74 : : \* pattern  75 : : \* Parameters:  76 : : \* code : uint32\_t [IN] [pattern code]  77 : : \* Return Values:  78 : : \* X component of the K vector of the pattern  79 : : \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  80 : : int32\_t getKX(uint32\_t code) const throw()  81 : 999 : {  82 : : return PatternEdgeVectors::mEdgeVectorRepo[code].mK.getX();  83 : : }  84 : : /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  85 : : \* Function Name: getKY  86 : : \* Purpose: Get the Y component of the K vector of a given  87 : : \* pattern  88 : : \* Parameters:  89 : : \* code : uint32\_t [IN] [pattern code]  90 : : \* Return Values:  91 : : \* Y component of the K vector of the pattern  92 : : \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  93 : : int32\_t getKY(uint32\_t code) const throw()  94 : 5553 : {  95 : : return PatternEdgeVectors::mEdgeVectorRepo[code].mK.getY();  96 : : }  97 : : //Constructor: assigns the vectors  98 : : PatternEdgeVectors() throw()  99 : 1142 : {  100 : : }  101 : :   102 : : } PatternEdgeVectors;  103 : :   104 : : }  105 : : #endif //\_PATTERNEDGEVECTORS\_H  106 : : |

|  |
| --- |
| Generated by: [LCOV version 1.9](http://ltp.sourceforge.net/coverage/lcov.php) |