```
/*
 * imx219.c - imx219 sensor driver
 *
 * Copyright (c) 2015, NVIDIA CORPORATION, All Rights Reserved.
 *
 * This program is free software; you can redistribute it and/or modify it
 * under the terms and conditions of the GNU General Public License,
 * version 2, as published by the Free Software Foundation.
 *
 * This program is distributed in the hope it will be useful, but WITHOUT
 * ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
 * FITNESS FOR A PARTICULAR PURPOSE.  See the GNU General Public License for
 * more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program.  If not, see <http://www.gnu.org/licenses/>.
 */

#include <media/camera_common.h>

#ifndef __IMX219_I2C_TABLES__
#define __IMX219_I2C_TABLES__

#define IMX219_TABLE_WAIT_MS  0
#define IMX219_TABLE_END  1
#define IMX219_MAX_RETRIES  3
#define IMX219_WAIT_MS    3

#define imx219_reg struct reg_8

static imx219_reg imx219_start[] = {
  { 0x0100, 0x01 }, /* mode select streaming on */
  { IMX219_TABLE_END, 0x00 }
};

static imx219_reg imx219_stop[] = {
  { 0x0100, 0x00 }, /* mode select streaming off */
  { IMX219_TABLE_END, 0x00 }
};

/* 0.912 GHz */
static imx219_reg mode_1920x1080[] = {

  {IMX219_TABLE_WAIT_MS, 10},

  /* analogue gain setting */
  {0x0157, 0x00},
  /* 0x0158, 0x159 Digital gain */

  /* Course Time */
  {0x015A, 0x04},
  {0x015B, 0x22},

  /* Frame Length: 1766 */
  {0x0160, 0x06},
  {0x0161, 0xE6},

  /* line length: 3448 */
  {0x0162, 0x0D},
  {0x0163, 0x78},




  /* Image Width: 2599 - 680 = 1920 - 1 */

  /* crop_rect.left: 680 Offset */
  {0x0164, 0x02},
  {0x0165, 0xA8},
```

```
  /* crop_rect.width - 1: 2599 */
  {0x0166, 0x0A},
  {0x0167, 0x27},




  /* Image Height: 1771 - 692 = 1080 - 1 */

  /* crop_rect.top: 692 */
  {0x0168, 0x02},
  {0x0169, 0xB4},

  /* crop_rect.height - 1: 1771 */
  {0x016A, 0x06},
  {0x016B, 0xEB},


  /* crop_rect.width: 1920 */
  {0x016C, 0x07},
  {0x016D, 0x80},

  /* crop_rect.height: 1080 */
  {0x016E, 0x04},
  {0x016F, 0x38},


  /* X odd increment: 1 */
  {0x0170, 0x01},

  /* Y odd increment: 1 */
  {0x0171, 0x01},

  /* Binning Mode Horizontal: 0 */
  {0x0174, 0x00},

  /* Binning Mode Horizontal: 1 */
  {0x0175, 0x00}
};

static imx219_reg mode_1280x720[] = {
  {IMX219_TABLE_WAIT_MS, 10},

  /* Course Time */
  {0x015A, 0x04},
  {0x015B, 0x22},

  /* Frame Length: 1766 */
  {0x0160, 0x06},
  {0x0161, 0xE3},

  /* Line Length: 3448 */
  {0x0162, 0x0D},
  {0x0163, 0x78},

  /* Image Width: 2919 - 360 = 2560 - 1 (2 X 1280) */

  /* crop_rect.left: 360 */
  {0x0164, 0x01},
  {0x0165, 0x68},

  /* crop_rect.width: 2919 */
  {0x0166, 0x0B},
  {0x0167, 0x67},

  /* Image Height: 1953 - 512 = 1442 - 1 (2 X 720) */

  /* crop_rect.top: 512 */
  {0x0168, 0x02},
  {0x0169, 0x00},
```

```
  /* crop_rect.height: 1953 */
  {0x016A, 0x07},
  {0x016B, 0xA1},

  /* image width = 1280 */
  {0x016C, 0x05},
  {0x016D, 0x00},

  /* image height = 720 */
  {0x016E, 0x02},
  {0x016F, 0xD0},

  /* X odd increment */
  {0x0170, 0x01},

  /* Y odd increment */
  {0x0171, 0x01},

  /* Binning Mode: X (2) */
  {0x0174, 0x01},

  /* Binning Mode: Y (2) */
  {0x0175, 0x01}
};

static imx219_reg mode_640x480[] = {
  {IMX219_TABLE_WAIT_MS, 10},

  /* Course Time */
  {0x015A, 0x04},
  {0x015B, 0x22},

  /* Frame Length: 1763 */
  {0x0160, 0x06},
  {0x0161, 0xE3},

  /* Line Length: 3448 */
  {0x0162, 0x0D},
  {0x0163, 0x78},

  /* Image Width: 3279 - 0 = 3280 - 1 (2 X 1640) */

  /* crop_rect.left: 0 */
  {0x0164, 0x00},
  {0x0165, 0x00},

  /* crop_rect.width: 3279 */
  {0x0166, 0x0C},
  {0x0167, 0xCF},

  /* Image Height: 2463 - 0 = 2464 - 1 (2 X 1232) */

  /* crop_rect.top: 0 */
  {0x0168, 0x00},
  {0x0169, 0x00},

  /* crop_rect.height: 2463 */
  {0x016A, 0x09},
  {0x016B, 0x9F},

  /* image width = 1640 */
  {0x016C, 0x06},
  {0x016D, 0x68},

  /* image height = 1232 */
  {0x016E, 0x04},
  {0x016F, 0xD0},

  /* X odd increment */
```

```
  {0x0170, 0x01},

  /* Y odd increment */
  {0x0171, 0x01},

  /* Binning Mode: X (2) */
  {0x0174, 0x01},

  /* Binning Mode: Y (2) */
  {0x0175, 0x01}
};

static imx219_reg mode_table_common[] = {
  /* software reset */
  {0x0103, 0x01},

  /* Access Code to registers over 0x3000 */
  {0x30EB, 0x05},
  {0x30EB, 0x0C},
  {0x300A, 0xFF},
  {0x300B, 0xFF},
  {0x30EB, 0x05},
  {0x30EB, 0x09},

  /* number of csi lanes = 2 (1 + 1) */
  {0x0114, 0x01},

  /* CSI data format */
  {0x018C, 0x0A},
  {0x018D, 0x0A},

  /* dphy control */
  {0x0128, 0x00},

  /* external clock frequency = 24MHz ?? */
  {0x012A, 0x18},
  {0x012B, 0x00},

  /* analogue gain setting */
  {0x0157, 0x00},
  /* 0x0158, 0x159 Digital gain */

  //Clock Settings

  /* VTPXCK_DIV */
  {0x0301, 0x05},
  /* VTSYS_DIV = 1 (Divide by 1 I think)*/
  {0x0303, 0x01},
  /* PREPLLCK_VT_DIV */
  {0x0304, 0x03},
  /* PREPLLCK_OP_DIV */
  {0x0305, 0x03},
  /* PLL_VT_MPY = 57*/
  {0x0306, 0x00},
  {0x0307, 0x39},
  /* OPPXCK_DIV */
  {0x0309, 0x0A},
  /* OPSYSCK_DIV = 1 (Divide by 1 I think) */
  {0x030B, 0x01},
  /* PLL_OP_MPY = 114*/
  {0x030C, 0x00},
  {0x030D, 0x72},



  /* CIS tunning */
  {0x455E, 0x00},
  {0x471E, 0x4B},
  {0x4767, 0x0F},
  {0x4750, 0x14},
```

```
  {0x4540, 0x00},
  {0x47B4, 0x14},
  {0x4713, 0x30},
  {0x478B, 0x10},
  {0x478F, 0x10},
  {0x4793, 0x10},
  {0x4797, 0x0E},
  {0x479B, 0x0E}
};

enum {
  IMX219_MODE_1920X1080,
  IMX219_MODE_1280X720,
  IMX219_MODE_640X480,

  IMX219_MODE_COMMON,
  IMX219_MODE_START_STREAM,
  IMX219_MODE_STOP_STREAM,
  IMX219_MODE_TEST_PATTERN,
};

static imx219_reg *mode_table[] = {
  [IMX219_MODE_1920X1080]     = mode_1920x1080,
  [IMX219_MODE_1280X720]      = mode_1280x720,
  [IMX219_MODE_640X480]       = mode_640x480,

  [IMX219_MODE_COMMON]        = mode_table_common,
  [IMX219_MODE_START_STREAM]  = imx219_start,
  [IMX219_MODE_STOP_STREAM]   = imx219_stop,
};

static const int imx219_30fps[] = {
  30,
  10,
  1,
};

static const int imx219_60fps[] = {
  60,
  30,
  10,
  1,
};

/*
camera_common_frmfmt: {struct v4l2_frmsize_discrete size}, {const int *framerates}, {num_framerates},
{hdr_en}, {int  mode}
*/

static const struct camera_common_frmfmt imx219_frmfmt[] = {
// Frame Size,    Framerates??,   Num Framerates, HDR Enable,  Mode
  {{1920,1080},   imx219_30fps,   3,              0,           IMX219_MODE_1920X1080     },
  {{1280,720},    imx219_60fps,   4,              0,           IMX219_MODE_1280X720      },
  {{640,480},     imx219_60fps,   4,              0,           IMX219_MODE_640X480       },
};
#endif  /* __IMX219_I2C_TABLES__ */
```