|  |  |  |  |
| --- | --- | --- | --- |
| Team name: | *B7* | | |
| Homework number: | *HW08* | | |
| Due date: | November 23rd | | |
|  |  |  |  |
| Contribution | NO | Partial | Full |
| 1 *Giampà Simone* |  |  | *x* |
| 2 *Massa Giacomo* |  |  | *x* |
| 3 *Raduzzi Lucafrancesco* |  |  | *x* |
| 4 *Micelli Johanna* |  |  | *x* |
| 5 *Galimberti Claudio* |  |  | *x* |
| Notes: | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Project name | HW08: LED matrix | | |
| Not done | Partially done  (major problems) | Partially done  (minor problems) | Successfully completed |
|  |  |  | *x* |
| SET UP  First of all, we configured the SPI by setting the pins PA5, PA7 and PB5 respectively as SPI1\_SCK, SPI1\_MOSI and GPIO\_Output. We also configured TIM2 as an internal clock source and we enabled the global interrupt in the NVIC settings, imposing values to the ARR and the PSC so that the interrupt is generated every 4ms.  CODE  We defined the 5x2 matrix containing the content of the row and the column of the letter that we want to transmit, corresponding to the position of the LEDs that we want switched on. We started the TIM2 generation in interrupt mode in the main loop and then the rest of the code is executed in the callback of the timer. Every 4ms, using the HAL\_SPI\_Transmit and the HAL\_GPIO\_WritePin functions, the information of a single column is transmitted to the LED matrix. In order to transmit the data to the other columns, we simply used a counter variable that we increase at every callback execution, so that at the following callback we are able to transmit the following column. | | | |
| Professor comments: | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Project name | HW08: LED matrix with DMA transmission | | |
| Not done | Partially done  (major problems) | Partially done  (minor problems) | Successfully completed |
|  |  |  | *x* |
| SET UP  Starting from the previous setup, we enabled the DMA transmission mode for SPI and its interrupt.  CODE  We defined two 5x2 matrices, one corresponding to the pixel mappings for the letter B and the number 7. The approach is the same as described above.  This time, in order to transmit the two letters, we exploited a second counter variable that we called “elapsed”, that we initialized at zero. Every time TIM2 generates the interrupt the elapsed variable is increased by one. Since every letter needs to be displayed for one second, and the interrupts arrive every 4 ms, then the letters are switched every 200 callbacks executions. This means that the first letter is transmitted to the LED matrix when elapsed is between 0 and 200, and the second letter when elapsed is between 200 and 400. When this counter reaches 400, we reset the counter to zero. | | | |
| Professor comments: | | | |