**Daniele Cicala**

**CODICE PERSONA: 10630561**

**N° MATRICOLA: 910392**

**INDICE**

1. INTRODUZIONE 2

1.1 PROGETTO 2

1.2 ALGORITMO DI EQUALIZZAZIONE 2

1.3 MEMORIA 3

1.4 COMPONENTE 4

2. ARCHITETTURA 5

2.1 FSM 5

2.2 STATI DELLA MACCHINA 6

2.3 DESIGN PROGETTUALE 8

2.5 CONTATORI 9

2.6 CALC\_SHIFT\_VALUE 9

2.7 SHIFT\_LEVEL 10

3. RISULTATI 11

3.1 SINTESI 11

3.2 SIMULAZIONI 11

4. CONCLUSIONI 16

**1. INTRODUZIONE**

**1.1 PROGETTO**

Il progetto si basa sull’implementazione di un componente hardware adibito all’equalizzazione di

un’immagine, tramite la lettura di ogni pixel che la compone e la conseguente riscrittura del valore

equalizzato. ![Immagine che contiene testo, schermo

Descrizione generata automaticamente](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RD4RXhpZgAATU0AKgAAAAgABAE7AAIAAAAPAAAISodpAAQAAAABAAAIWpydAAEAAAAeAAAQ0uocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAERhbmllbGUgQ2ljYWxhAAAABZADAAIAAAAUAAAQqJAEAAIAAAAUAAAQvJKRAAIAAAADOTUAAJKSAAIAAAADOTUAAOocAAcAAAgMAAAInAAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAADIwMjE6MTI6MDEgMTI6MjM6NTMAMjAyMToxMjowMSAxMjoyMzo1MwAAAEQAYQBuAGkAZQBsAGUAIABDAGkAYwBhAGwAYQAAAP/hCyFodHRwOi8vbnMuYWRvYmUuY29tL3hhcC8xLjAvADw/eHBhY2tldCBiZWdpbj0n77u/JyBpZD0nVzVNME1wQ2VoaUh6cmVTek5UY3prYzlkJz8+DQo8eDp4bXBtZXRhIHhtbG5zOng9ImFkb2JlOm5zOm1ldGEvIj48cmRmOlJERiB4bWxuczpyZGY9Imh0dHA6Ly93d3cudzMub3JnLzE5OTkvMDIvMjItcmRmLXN5bnRheC1ucyMiPjxyZGY6RGVzY3JpcHRpb24gcmRmOmFib3V0PSJ1dWlkOmZhZjViZGQ1LWJhM2QtMTFkYS1hZDMxLWQzM2Q3NTE4MmYxYiIgeG1sbnM6ZGM9Imh0dHA6Ly9wdXJsLm9yZy9kYy9lbGVtZW50cy8xLjEvIi8+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczp4bXA9Imh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8iPjx4bXA6Q3JlYXRlRGF0ZT4yMDIxLTEyLTAxVDEyOjIzOjUzLjk0NzwveG1wOkNyZWF0ZURhdGU+PC9yZGY6RGVzY3JpcHRpb24+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iPjxkYzpjcmVhdG9yPjxyZGY6U2VxIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpsaT5EYW5pZWxlIENpY2FsYTwvcmRmOmxpPjwvcmRmOlNlcT4NCgkJCTwvZGM6Y3JlYXRvcj48L3JkZjpEZXNjcmlwdGlvbj48L3JkZjpSREY+PC94OnhtcG1ldGE+DQogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgIDw/eHBhY2tldCBlbmQ9J3cnPz7/2wBDAAcFBQYFBAcGBQYIBwcIChELCgkJChUPEAwRGBUaGRgVGBcbHichGx0lHRcYIi4iJSgpKywrGiAvMy8qMicqKyr/2wBDAQcICAoJChQLCxQqHBgcKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKir/wAARCAD+A6MDASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3+Pn6/9oADAMBAAIRAxEAPwD6RooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACikJA60bh60ALRSb19aN6+tAC0Um9fWjevrQAtFJvX1o3r60ALRSb19aN6+tAC0Um9fWjePWgBaKTevrRvX1oAWik3r60b19aAFopN6+tG9fWgBaKTePWjevrQAtFJvX1o3D1oAWik3r60b19aAFopN6+tG9fWgBaKTevrRvX1oAWik3r60b19aAFopN6+tG9fWgBaKTevrRvHrQAtFJvX1o3D1oAWik3r60oIPSgAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooA5bx7f3en6PDNYzvC/m7SyntXn/APwk2u4YjU5yD05H+Fd18RmI0SDHebB/KvMxleV6+lMRsxeJ9aON+oTH8akPiLWeWXUZvpnpWQpDKD0zSg7cdT70XGaZ8S6zg41Gb65qk/i/XYrsZ1GZkUjIz19ahKk8p19PWoWkRCNyjJOSCKLgdFN4t1YJmN36dd5NZ3/CVa+QT9vlHPHNVdxHMbdOuKkhuY+k0akZ546/Wi4FuHxNrvmBXvpWLfd+YVO3iPWo22yXsg9y4z/Ks6aCG5VTbssRHYdDTRBIq/MM7ec7s0XAu3XibWoo1dNSmwSR94H+lRjxVrZTjUZ/zFZp2mM+YGBJ6EVGsgTgKMHrRcDai8VawMeZqE5ycdatweJtSkt2kN/J8oPG6ua3b5lQA/ezU12Ba26oFBdidze2elFwOkGu6q8ny6hLgDBXPeoLvxHrFsqkX0p3HH3s1Q0+ZJLcPcNjIx0x0qSVLSYY5/Gi4FqPxTqZQbryXP8AvVZh8TX79bx/oWrmrwLbOAjHDDNNQuiiU5AI4OKLgdgPEV33u3P0c1Mus3bqMX0yn3auPS6xnjd7VMLkHktt9s0XA63+0tRP3byUj1D1FNqGptGfKv51fHBDVz9veurAxPwOtaa30uBlV+tFwIl1TxBsYG/uMg9SR+lF1rutW0S772fLD72elWFvN/DqB7+lNlQypkbSDRcCrbeItWYkvqUr8ZxnpUp8T3quUbUpA3oTUJ04OrAqEJ7jismfTZY5mJ+YepouBsp4qvjAHe/mySRgGmT+JtU3xCLUJQG6kmsDaDwcgDuKQ7TjG7OaLgdBdeIdaV/3N9Lt25yDyTVY+KNdDKPt8u7H3SetUrVpPtG9nZlxjHWre2OWYSfxqMZz0ouBd/4SnUxZ7jeSl8cjdVePxVq+477+XBHHPSq4WOG1KS/dJ6/jUAgiEwKsGXONvSi4Gla+ItcaciW/mdAOCDU8XibVpFYrcSMFPzHeBgVQ82OCLa7BR9arwwERy7HDCTvRcDZTxDrU3MFw7Keh8zrUi6r4gZfmuyp/381mWTmyt2Eg3tnPA6+1Xlu8oCQE470XAdJq3iBRzfMPoc16TpDvJpFs8rFnaNSzHucV5PqWpvbohh2sWPPGa9V0Ni+h2bnq0Kn9BQBNJOySMBtwBuJZsAVXbVIFbDXdoD6GcUt50nz/AM8jXgmp6HaQ+I5RqcQKRXjyW1wp3EEOSAfyAoEe9rqMbfduLY/Sal+3oG2me3z6edXjmh6U2nF3Egfe5fco4Ge4/wAK6eKa1v23HYbhFCebjkcdD3x3FFwsd79uX/ntb/8Af2mjUomk2C5tS/8AdEwz+VecXMh3yWUwZfNQrt3EB1IwdrCvMoPB3iW2+KUXiLSwYoBe7H3SjesQUDPPUEZHfnFFwsfS5vNoy0sAHqZaY2pRL965th9Zq8y8S6frOv2Elvb+IW+bpC48vcOuODg/pXLv8PdRlMAvJNQmt4l2swUNISSW6dQee4NFwse7/bl4/fW/PT971pDfopw09uPrNXjPhzUrR45tLutfuGubYlIdsaLhRwAWZThuK155oDaGK+muH81XVJcoXbg4OQvTPYUXCx6cNSiPS5tj/wBtqBqMZ6XFsf8AttXzt4astVj1GSykidHJYCYj7+Bxyex461p3E154ckuLhkj3MoIAQEH5unt6/hRcLHvP2z/prB/39o+2c482DP8A11rzTQfGskmn6tPc27EW7ny5f4FygYLk9zmuv0x7bUdl8FDN5KqrrIW+U9VPbORRcLG59sH/AD1g/wC/tH2v/ppD/wB/K8+8aLqUV9YvpX2pYFJ87ykOB7nHX8aIPGZEaAyKrZ2BGPzNjvii4WPQvtR/56Q/9/KPtR/56Q/9/K57SfFVtf2azISpyQyu+CMcHj04rYTUFHzNnYB1PNFwsWftR/56Q/8AfykN3jrJD/38pq3tu7Y3Yqfy0kXsymi4WIBqEZJAuLYkdQJulL9tG3d50GPXzayR4O06K6uJoYlj88fOEUA/mOai1bQrgaQlrpJjVFJEkTjh1x90HnFFwsba3yt92a3P0lpftf8A00h/7+Vx2lD/AIR61uZL2ze1TI3SA7geO2OwHAzXN3vxPnvI9StNL0598RKQ3CvkZB7/AIelFwseq/a/+mkP/fyj7aCSBLBkdf3teZp8TSNSgtpYIo0SIGfGXYnHPPbBqno3inTzq2tNeXbWRuoz5czycLnIGB696LhY9WN+gJBntwR1HnU4XmRkSwEe0teOTeD7ttIk1a11KS/t/seNgc+ZP1J5688cdeMVo+BodW0u/jhleZjfWYmS2mlLCDaQMZPQ89KLhY9R+2gEAzQAnoPNpBfoW2ie3J9PN5rxu91DWLTxI007zNMt0BCpJ5XdgDHoRntzW/ovhmWbxguvxyPAY2fz7dnyrMy8kDt1ouFj0f7X/wBNIf8Av5R9r/6aQ/8AfyvO7HxVLPd6yZNQlii0xgzIqoVfPQFiCRyO1a+jQTeKNJGoRa7fQxzE/JAY8Jg9AdvTtRcLHW/a/wDppD/38pRdFmAV4ST0Ak/+tWMnheMW6xzanqUpA++bkgn8sUltoVnY6vDOlxPJOvG2W4ZuCuM7c47dcetAHTUUUUhhRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQByPxEx/YkOf+evH5V5nntXo/xMBOg2+P+e/9DXmcE/mkoQQy+tAE4JVeOaeGz6VExIOaUENyOD/OgCdSM9OetNdBI2GUYHIb1pqvnhvzp4OM5HB6EUAR5MXAHA96dsSUZHB9qeQM+q/yqN4zH8ycr3xQA0pIjeo9RTkuXTrn8acJRs55pflYep9DQBILhJOJEB/CkEFo5wRs9cHFQEBe2KM564oAnfT4T80Upz271C1k8kh+0sSh6MnOKRZSnqKlF25/CgB8cMEcIi81WUZxx/WnxHyBtDAj6nNMH2ecHzk2Mf4lpDYSbSYZldfQ8GgCHUVNyymLPygjmpLdnW2WN88ZGMVVlkdH8p+Cvar+kmOWGYyvtCEMSewI/wDrUAUbvCsuwY45xUSlmzjmr891HJIPLbKYXjH+1zToQh1KaAhdqlsH8qAM6CZlzyeauWmoG3kJkLSK3XnpSXyLFN5UagBlBzinf2W8lsrjeGK5C4oA17e5tbnPlyMCOxFaMSBIwAcg85rlhbvbBfM3KcdcYxU66rcpIo3ExCgDoyVzjNRXnyWrMmM9DmorC4W9hLgEFTgg1PLB50bRkcEHrQBRbRwbcSRj94wyVzx+dQrYnaRcWxBHoc1swxGKFE3MQoxVXU7VrmEBC+R6HGaAMaeOG2J2kqcZwT0qCIvNJiOQOVHQjmrcunvIMzRv19eaItPWGTMRkH4igCB5GG4lA+Dj1waovDMQ5GCS3Y1rPZhCcOgLHJxnNVn0y5x+6dGXOTzQBmOZM5kBye5FSwahLCMLgj0xU8kN7Hncm4dsngVE1uDx5DLk9jQBeh1aGXO9dmBmpFvrVz97nP51mmxLMURW3A89MVE9pKhPy8fSgDbaa0fckuw4OD2r1fRcf2JabPu+UuPpivBXZhMQw+or3jQP+Rfsf+uCf+gigAvOk/8A1yNeOXNi8TNDdOFkGUlU8ozdDz255r2O96T/APXI14PbeJo9b1i9ScLbT/aJPLXcMSIWOOM9cYpsSNvQN1qJrdpNuGyqMM49s1ssVkAkgdUZe4xz9a52SdbcFJFLRn7ykfhV2OYwLHLA/mwhcdCzJ7MO49+tIZvSWq6hB9muYmYZ3CRW5U+qkdD7fzrltXt/EGm6zY3NnILi1gBBGcFh6HPAJHSuisLpHAe3OJNvMRPX3B71PbRxSS3UsaoZJsCRSeGIH93seaAKDE6nbq/EFySeNvHXjPcfypsXibUtJYW15EAi8fvDuU/Q1FeaLKCZNMkMc1v90A5A/wBlh3HvU9nqq3aLb6rCiuVwyn5gfp7UASrqHh3UbwXNzZrb3ZIJmt2KFj74xn8c1ZvfDaalcxXWmamVCIR5cvz5J6nJ6HmqN14Z06Vs2xeNuoER4H4VUGmXljIBBeb1HY8EUAZOs+FPFSzs8dktwh4MttMG/wDHeDXO6/a+IxDEbm11BFXCSiSI8rnjnHSvSLbWrqIhTcfvR2JyK0rXxY0rmK8Ee0cE/doA8x8dalYX/hiPTfC8MzXDbLm7igGSTgLk46424rr/AIYXOpaX8PrZlsXkmXzPMhY4kA3ZGF/E+9bsttoM1wmoWhWyvFYnz7dVB/HqD170t5e+JYxJcacLTUrdSdohXa5HHVSfr0NAGzpepz6rYxyytGrSLkxMnzJ7HmvDvHGrvafFuWzmcrAs0SsEGMKyr/jXbzePlg/dCwmtrxXCvC8PlcHPIPGehrF13UvDuo6t513pi3k8fPnM21sgZH17j2xQB0OreFLyGN4tCuIWhZSCrDaVz71z8HjxrW5ke6nnLRnY8X8BPP8AhXpVlaJcaJb6k0zrHLbLP0zjK7sZ714zfeFtI8ufUU1a7S086GJg0Y3ZkG9WznpwaAOut/iZv1eGzuI1RJeiKMk/j2rttP8AGekyahBpqyyedL91iuFHGeT2rznV/hpAy2PiBNTWG1jhQEhTl9x4P1+YflWFBa6rqFncNognv0tGWMLCm5uBwTkjnFAH0WpbOQTzUoUSKSRzXjHgDxP4hPN89xIlu7wyQSYCx4OMZPVsmvR28b6XBf2un3TtFeXONsW3OMnvQBvmFCpDLkEc5rkdX8EaYkM09nbRJcyMGDcgk5ycY6muvjlEyB42DKe4OaqamLllga1jjkKSrvRx1XODj3HWgDzQeEdZ8gRwzGOVmLbrmPeU9gwxx+FQ3Pg7WUkkm1CyhvlmwriIDkY6sNuf516zd2FtqCKt3AsgU5AasS9vby3u57OCw8uNISYrhpRtLZxjHXp3oAwIdasvDmnQ2kk62mFLCGRcZXPOAPrWtb6nZ3axXgEf2jaUQ78kA4OMn8K4nU7fxvc6yfKtENgv3nj2SEAc42kjOfwrpLqzuWSwIsIIYWRjdJIWVtxAxjaMf4UAYWveLhY6jFpurwWlzMQ0heJuYSASCDjg4H6iqV18YrVLSNdJtTNMyDe0sp2o/wDdqLW2sNLnmbU9GaaCKALHLLl2fPYEdAM4z71xk1p4MmQRw3F5pTCEzsxw6Rg87eeSenv2oA9O8L+JfDms2cgvn0+C9uwftManCydBznr1712EGjJY6WljosgsYUYMgiQEDnJHPHP9a+apfAuqxwu1pe2t7C4JR45QrFTjHynnOOSK0tO1bxn4bVlhubny7tgI/M+cAJ1Kjnr/ACFAH0oJJRgbF+pNcbF4W1VPi4PEjXStYyx+U0A/gwmAOvPOT+NeTaP8XPElpJEt1dpcx7XG2VNpyVwCT7EV03ww+KmseLvGNlpV4YTAqSM7bcO5VT17e/5UAe9UUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHJ/EJFfRoAwyPN7/SvNpbVGO9flcd69N8d86VAD/wA9f6V53ImDx070AUmBHDjDUzB+lXtpdcHDLjjjkVBLCY/cetAEKtj/AApyOUPGSp/hpMUhFAE4cHvTwxzkc1TDeozUyE7flOR6GgB7xo4JTjPUDtTgoK4HUVHGyhjggZ65p7ZUZH3fWgBpbseaZtJ6U/cp6/nQcHoelADBx1FRuMyJs6Z5qTOTSHmgBjs0QJPIqSC7b7ynAqKZXMZAwagLeXCoK/MOM+lAGqJ4rhMSAEe/aokaAW9zGg8stwST1HUVTgJZGwcHNQuZEfGe+DQA3y5dquilhnqO1btrKZLCW5iTMqE7mLDP5YqppV79lkkUqHjkGce9R2cpjvZrZjiO4J4z9cUAaskd1qOnwzQqN2c53Yz+lWoLuaE28Fwjoz8FuopbW6S0sxEsBbYMDaetW7a9trpiqqVdRkhlxigCZ7dZkKyneCORisu70fYrPbnIxxGR3rVluYbePdLIFHbNMgv7a6GYpVP1ODQBzK3UlqTnMTjqDwa1dJ1d7t3jkXO0ZDYrQns7TUI8yosg6bhWfNoYh+axk8oKOQSefxoA1PPycY59M01rjacbGrmTHqizB0jkKr0dfmrWt9YjSJRd/u5RwQR1oAtvMzHCDB+lRkTH1/4EvFW47mKRdynK+tVZdasI3KPKCR/s0AZdzfLb3HlTgMcc4qP7da4Jzs46Y61qqNLvQXZY2LD+M803+xdNdiViB+jUAZrTAx5SRipFRlzIPkOSOuK0JdDtWPyNJHn+6aZHoqQbjC5y39/mgCokcnO5WI9TTwAHCMGVj0+brVn7PNHhS6gD0FNZZOu5d3qRQBG8CK3zxs2evAr1fRcDRbTAwPKXj8K8pM0o4bB/CvVtF/5Atp/1yX+QpgNvek//AFyNeHeJPhCmozNdWGoGK/tpCYZVjA3KDkKwHXHqOfrXuN792f8A65Gsa8gK5khHOSdvrQxI8xh8+VPI1S3FreJ8rLn5ZP8AaU+h9OtVUMtnM3ksxIOCB2ru72yW/QRz24mUn76sBJGefXqOOnWue1HRprRVMoYrjKyAYJ/z6GkMpWeobGLkbWJJ3Ieh9a0NP1eKSRhqKRW0+75LmJ8LL6Z9D9ayfsyqwLYU4wpA4/SqrN5QZgUbHXK9fqKAO+WbfIrhvm6hlPP4HuPaq17p1tqMoeJvsd4PuyIoIY+4rkrPVzvREnMDMMKM5iY+hHb8K3Li8lntWt5i1q8kZVLlOQCR1VqAHw2N9FqhF0xVNmNyvxvz156VdncFfLu13MvRwMGnQyy3Nukjf60Lgjrux3+vemS3UM2BO5U4wCOooAybmKQNmJi6jk89KrbI5mxOxTd1YEq39RW19lVULRyK7EYDpwfyrPn2tKFDLvXqB/F+FAGNq082izWC2kqzC7uVgx0PPt3q/b+JW0iVvPZrYqM5ZtopksIcAx5WQHIyf8a5vx5JfXXhaa3ltZZyMNH5K5bgg9aAPTrHxfpurW4i1GOCeNxw5UMpHrXJ6/8ADTTjaT6jo+rGC3DeeyzfMIgOuxhz+BzXnFzrEGj+F9MitmuEnmhKSQbf9WQOSD6810R1vVrjwGs2nq0kckRSQEjdt6Ekdxx/jQBc+Id74h8P+CvC9tZ3n2i0hTaZYuFlI5j/AA2dq5vwTrK+KIbrwtqcwgmvLm38q5HBygbCkHg8dwPrWQ3iLVNbhjtWXbJpak2zhwNrdgMnqa6/x1/Y6+EfDnizw5pwtNSW7hnvTFEQxwpJLH2K0Aeq3aaTqHh6fwfJdTNPaWcZYRA7iExjAA7lf1qh8M5tJ0/TdQWbdbyq6mU3JwcBcDkgeh461e8JW0eseIU8cx3LfZ9UsEWO32D92QeckdeQa6S5u9A11ptJvJI5HYmNoJgVZu3APJ78igCafw/p91bYhi8oM5lDRHGWPJPvzzXG6r4ONtL9puVaR44nYXin/VjuPbjPXPevR4Yo7W3SNMLHGoUZPQCmJc216jpHJHMB8rhWDAexoA4Tw34isNG0tUjLPBIfMySSFyoPUk9+Prmuv0XXtO122abT51kCHa4B5U+hrmPE3wt0rxHZPHYXEulF+GFrjy2+q9PyxVHSfDeqfD/T5otJiju7cxl5pggEjuAADt78daAPR9y+tRSxwyHLqCR3ry3wt8VrzVNWa31LSp7aBn2B5I2Uk+wPUcGvTYbyznk8uKWJ5MbiquCQD6igBrQQDdsiADdccZqveXUFnAWuWaCHgFycDmrlzdWdnHvupYoU9ZGA/nWPqtno3jLTH077cs0JIZvss43DjjkdO9AEiXVtK0aw3kEwbohIJbv361m6lpmmXqM13pljcjJwzBTz+INYVz8G4Gniaz8Q6jbxRHKx/K2OAODj0FTR/DrVNIs1g0DxBIAFbcl2CyszHJPHSgBtx4a0BIYbiHSYA6vvUoxXa3TI2/QflVuw03T/ALRc3AsoUkuMmQHLA56kemfaofDfhDxJpN1eT6xrCagJgDFEmQEPtntXTpFMoAa2QYHPIyfxoA5W58FaBqaMH0+3YMRueOVlKkexrO8DfCjRfB3iSLUtPu7iacbl2zSK2NwOcBQP8iu2SA2ly91Darvl++dx5/zmpoZ0mvYf3Gx9x+YDpx60Ab1FFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQBynxAbbo8B/6a/0rgA4dcjGa774hHGiwZ/56/wBK82ztYFTQBbPHb8qBzwTx6Goo5s1MB3HIoAhktgRujGfUelV2Q54rQV8e9JJGsnK9aAMwpn2puSpzmrrR7fvCoZIe45FAEQdT94c4xmrMcgPHBB6iqTpjpTNxU8NigDRaBWX92cH0qBomB70xLtk96mW9RvvAigCNSR1/WlwDT22v900gXFADDG/8JyKiljWQYcYPY1ZxkZXGaguZzsAx82etAFfmGPCnv1oH7xWO7BNOZN8WRycdM1WQc88H0oAeGMTjOOO9PuQ4njlUHkgkjtUbEKo3HINWopFmgKHJIGDQBcgvzjG4MB71ZW+haTBba/1rnGZ16cD6Vait5PtAlVgwIzxQBtXCLfRhWmbjoc9KpS6fcWse6GQyeuBg/nSqsnp+tSK0qHjP50AO065v4sxhGx155xT9Qm1G4t9jDC542g80+HUpYG2LEcHkttp5ujIMOWoAz7a8vLGPyY3kAz029PpRPczXGPtA3/UdK0EljPBGTjvQQnUggfSgDMRtv3C6fRqrvZlpN/m5JOTmtOeVEUbYSxzjgVBHOJOHgaM9uKAK/k3KvlNrAdMtU1vcTwTmTDR8444zUrRI3IOPXmozEe0lAFiPxBtkBkeT0YEjH1rQttZt7txGu4NyeRxxWG8BPVUeiDfaS+ZFHtOMcc0AdK7Ajgg1Wd2zwpNYPnTszETyAsc4IqxHd+Uf9JkJBHBAwRQBoNIf7n6V6tov/IFtP+uS/wAhXkK3StgpI2D+levaKc6LaHr+6X+QpgNvOk//AFyNYS6hDPcSwM2GViAp4I5rdvOk/wD1yNefanMlxNLKrFHV2Kso6jPWhiR0zLjJ2depHegxiZfLlwUI5Vx1rAsNeYSCKc54/wBYOQa6CGSOcK0TqSw+6T1pDMHUfDA+aa0BZP4lH3h/jXOXWkyoS0Z3oByDwa9GWXy39R6VHdWEV1+8hwJO+BQB5Lc2jCMshxxypHWooNTmt42guts0DcGNvQ16Je6VEdyXEIXP8YHFchrnhoKjMFLr2dKAJ9Cv7KMq9mxRlXbskc569m/PiumksLPVf3hPk3DjlsfK35cV49eafc2zZWVwi/wqgJP44qxpfivUdJdfKnkkUHlJEyD+FAHe3vh7Ubdv3chyM7XU8EVQh+0RXGLwKcHq2Qfwplh8UmYBL3TmCnqyt+uDWw+t6TrFuGVhEx/vKAf/AK9AFFoYJ/70THoeo/AiqstnqcQ8yBhPFj2Y/wCNbCWMaR+YMnvkdP8A61OI8z97aOizf888D5h6UAea+LvClt4lVXEzWN3CSRwTGT3yO31FYl1r114fittITyorJLciQEBskk5Oe/1rd+Ifi6RdRsbZVjinXInP3SwyOvfsR3rlvFuirqelx6hog+07W2yZfLD2HTP/ANegCHV9MuTa2d3a3MF2ttEN8asAw6kdOoGeh5rT8PapP4h8G6voaJI7+WJEQMckjoB7e1claxLB+7luJoDvGQybcnp82Tnr6Vo+GfGVtoXiQTC3YRSHyrkJ/GM8Yz06A0Aeg/BzxrP4f8ODSdUmSLyZ3EdvcMEYKcdM89c/nXrUvizw6wt59Xkt7dxIDC8zDhvY187fE6GQa9aXunDY1zaiYkkdckYGemQAfxqj4b8J6j4u0iIf2okMtpcNiGUHgHB4PTFAH1Vq8I8QaNNYQ6o1v5o4liI3KR/MV5fL4I8beB9JmutD1NNXkJJMcMZVzk5JwSQSe565res9N1aKFN0kecAZMh/w61q20msWzA/eA9JOP1oAz/h54r8TzQz2Oq6DcQyJucPN8q9emcYJPXj3rN+JF14+1E2UWmwSadbsxWZrRmmwOu44APb/ADmuuh8T6nbap9kbRXFoYTI1yjKcyf3dv071rf2kZQN6yj64NAHnGm/EHW3t0017JryWCICSWaxZC/bkHofz75rOjltrfUmvIrO7064c5LWk5T/x0169FPHcqcROCCR8ykUy5hhCkva+Zg8jZkCgDx7xZpms+LNIjtLbxBNFEh3bbwfMx/3lBrAsfCni3w9pZTRmjvrmZ8u8dwikHnkbjk/kK9GvvEOrRak0dr4L820Vypk89A7AHqq/TtWljTL6NDcQyWby42xSna4PpwcUAcTZ+IfGVhe2ker6feWzxKZJHgkZoyMgDkZBOOcYrorv436TYagbe8nmjMEu2VRbq29fUHcMfkamufD69LTU7mH1BYn/AArC1Lwm93GBK+n3xHa5hH88f1oA9A8PfE/w14o1EWGlXkktw24geU20hepz0/OuqZsjg/lXg+j6Rf8Ag/UTfaNo0Ql2Mv8Ao1xuBB6/IT14qCCbxW9yzQ63rlmWmMrLcwpIoBOcDocUAe7SSdf3ZP4VDEUN3FiLa2/OcexrzG3+J2n+HW/s3xRfTy37Yk85bfyl2ngZAyB9a7fQ/EFtq2oxpbySMuQyEkEMCuc8HIoA7OiiigAooooAo63qB0jQNQ1IR+abO2kn8vdjftUtjPbOK8V/4aSm/wChYT/wOP8A8br17xn/AMiHr3/YNuP/AEW1fGNAHuUf7R80kip/wjCDccZ+3H/43R/w0fL/ANCyn/gcf/jdeIwf8fEf+8KSgD3D/ho+X/oWU/8AA4//ABun/wDDRcv/AELSf+Bp/wDiK8NHUVNQB7an7RMrOq/8I0gycf8AH6f/AIinf8NDy/8AQtJ/4Gn/AOIrxOH/AFyf7w/nUh6mgD2j/hoeX/oWk/8AA0//ABFO/wCGhZf+hbT/AMDT/wDEV4pUtAHsw/aElJA/4RtP/A0//EVG37RUqsR/wjScHH/H6f8A4ivHV+8PqKrP/rG+poA9q/4aLl/6FpP/AANP/wARS/8ADRM3/QtJ/wCBx/8AiK8Tp9AHtX/DRM3/AELSf+Bp/wDiKmb9oGVWI/4RxOD/AM/v/wBhXh46irkn+sb60Aex/wDDQcv/AELaf+Bv/wBhUnxF+PkvgPUtKtU8PJff2hpkV+WN4Y9m8sNv3DnG3r714sas/tDf8jH4W/7Fq1/9DkoA7H/href/AKE2P/wZH/41UqftYTvGH/4RCMZJGP7RP/xuvm2rcH/Huv8AvH+QoA+iP+Grp/8AoUI//Bif/jdXrD9p6a9V2PhSNNpx/wAf5Of/ACHXzXW3oX+qm/3hQB9Cf8NIy/8AQsJ/4HH/AON1esf2gJb2JnPh1E2nGPthP/sleA1t6H/x7S/7/wDSgD23/hekn/QAX/wLP/xFX9L+MMmo+bnRVj8vH/LznOc/7PtXi1b/AIZ6XP8AwH+tAHrX/Czn/wCgUv8A4Ef/AGNYviL42yaFHA40JZ/NJGDdlcY/4Aa52uN+IX/HtZf77fyFAHb/APDSMv8A0LCf+Bx/+N1LB+0VLMSP+EaQY/6fT/8AEV4VVux+8/0oA9x/4aDl/wChbT/wN/8AsK2fDvxkk12adDoiweUoORdbs5/4CPSvAa7H4e/8fd7/ALifzNAHtf8AwsBv+gcP+/3/ANjWroHiZtbvJITaiEIm/O/dnkD0HrXnFdV4C/5Ctx/1x/8AZhQB3lFFFABRRRQAUUUUAFFFFAHJ/EBd2jwD/pr/AErzaVWUGvSvH77NJtzjP77+lcAyq6ZXJ9qAM1JQG54+tWElI5Q0yeAEnAwfpVfLBsHigDTEyt94YPtTwD95DVCOTseasIccocY7UAWg6uNsg5qOSEc7T+VRiUbvnGD61OjDHXOfSgCm8J79PUVXeEitR4wx44J7iomjXHzcH1HegDKYFaQAmr7wr/Eaga2J+7+lAEKkjvUyz4+9zUTROv4UgPrQBbQo4+X5TUVzb+cMN8r9mHemA91qVJuzYI96AKOx7fduPUYGKb5n7vnnmtXy1dckAqapzWWCGhxjupoAqnY56fhUluUVXjVjvboSOlMC+VvMq4yeBU1tAJm8wbcehPQUAV4kd7gI3HPOa1ViCjKkY9qqXESW0m5eSc4p9rdF0PAzQBdAbqBT8HuMVAs7L0UU83BP8P60AP2ml2kf/rqP7Rn+H9aTzm7YFAEwDDvUglcelVTOfalM5A7D8KALDsX+8M/hULLj+FsexpnnOeh/IUxpX/2jQA4GME7Q2T1zSblB7/jTSzHuQaYUyeTQBMSo7UbwOgAqAnHU03cKALBdT1xUMkcUxG/PHTBozigtQAKkaDagNe16F/yAbP8A64r/ACFeKAivatC/5ANl/wBcU/kKYBe9J/8Arka4y/0tp7VHiwpKD5RxgEetdne/dn/65Guatr8xwiK6UbNo2sRQxI8t1S7vdD1PFxC/l7uGjUn/APXW5p3iOK5hHLwHGVZgRz9O1dNrtjDdQ4aITI3Q4zgV5d4i0mbRWDxrcS2rngxrkp7EZH6ZpDPWLPWjJEqznzB2kHetOGRJx5lrKSx6r3/KvIdC8XRW7JGzuqMcYdOjenNeh2M8V9DvRgj44YHigDokuUmPlzId3Y4qleaXG43xgqvXKrn8xVQX1xZMFvk81f76N/j1rWs76G7U+Q4bBwV7j8KAOP1Pw/HdISu3PdkGR+I6iuI1fwxcwqzJGx29GU5Fe0SQRhjKBsf+8BWbdWscj73hAbruUkE/XFAHgz2kiXAW6jZU6gg4U+2fSp7Y+Q+I5lznjIzj8s16tqHh+0vnxKhT2TBU/UEVzl38O4p2/wBEmUdSEOR+mCP1oAxrPxHf2rhYB56r1Ibdj8DzXVaVrNpq0Kx3sbW82OHGRXNHwhremSELYmZMYBiIOPy5qN/MRdt1E8EqdVYEY/QcUAaXjPwBb+KYU/tFsTxj9xqEK5b/AHXHcfrXk/iDw7q3hjSFtjFPPZRzmQ3MedpPAwcZHGK9W07X5rNtkxd4iMEbsr+FdnbLZX9jugKyow+cHsPegD5kvPEEzF3dFniSEIpK7ghbAPP0GKHa1jW3vpLHzkiPmxRj5coD13Y5Ax0JNeq+LPg7bXn2m40W5eyNwwd4wA0bH+lcnaeG/Ed54sTTLnTJhp9sATgGNJABj5WxjOO1AHN+PJItYuI9fspfLF5tWSD5vlbAxjjpj09K9m+H+kWVv4bsbc3HmzqgYkD77Hk/hWnrPheCy8Gzw2FjPe3MaM8BcAtnHHpn0ryLwJresS+P57DzWRXDB0Viu3HfB54x0NAH0Na27spjDgFexq6LZwo37Py61xsF7ervWC9MkkTbW2vkofT8qujUdQWLc9wzEcdqAOk+zMWHI+gzS/Zm28N0744rm01O+bhpyD6gCgTzLMZxMzNjGVOCKAOnjhlT7sxB9lqZXuu9wxH+6P8ACuRTUboyf66UDvul/wDr0rXUzOTIWI/vGSgDpL21NxHl2XcOhYY/Uc1z1wn2V1MsCykZIeO6Jwfox4qjc3BK4DAHu2wkCoFdmjHnzh85xtWgDQn1W2tjH9oihBcgDEisB74py6zatGrxiJg3+xiscJadGUsSckqDVaa2MjqLZD8zYAJNAHQnWUGCp2k+gpk2uK64ZWYHjBA5rk7h7m0kw8WQW+Yht38qF1JGb5H2NuwSwoAuahYaFqF0bm/0O1nmxjfIozgdK6Lwc8f/AAlFqsdqkOFYAqx4G3piuc/fSMrq8TDGSN2DXQ+C2f8A4Sa2DLgENyD7UAes0UUUAFFFFAGN4xGfA2ug99OuP/RbV8c/Z4/77f8AfNfY3jD/AJEfXP8AsHT/APotq+PaACC3jE6fO33h/DUeyP8AvN+VTw/69P8AeFV6AF2Rj+JvypfMj/2vyFNPSoqALcDxtcRj5uXA6e9aR05M/wCtb/vn/wCvWPbf8fcP++v866SgCn/ZqZ/1rf8AfP8A9erv9hp/z8H/AL4/+vSD7wrVoAzBoabh/pB/74/+vWa+nJ5jfvW6n+H/AOvXTDrWI/8ArG+poApf2cn/AD1b/vn/AOvUPlRj+JvyrSNZp6mgAEcefvN+VWJPL8xuW6+lVx1qWT/Wt9TQAv7v1b8q3/jfoS6nrPhmY3Bj2+HbZQAmc/NIfX3rna7j4tf8f3hr/sAW383oA8c/4RGP/n8b/v3/APXqO40SOxSNDcM+4k52Y9PeunrL1j70P0P9KAMX7DH/AM9X/wC+R/jU9vcxaYGUh5N5z2GKWqd999PoaAND+3Yf+eL/AJivS/hZ4eXxnp+oSR3JtPs0qqQybt2QfceleM179+zh/wAgjXP+u8f/AKCaAOk/4VQf+gsP+/H/ANetbRfhobQTf8TMNu2/8scevvXYVcsej/h/WgDl/wDhX5/6CA/79f8A1657xb8LDqcNsv8Aaoj2Mx/1Gc9P9qvVKztX+7F9TQB4r/wpU/8AQbH/AIDf/ZVS1H4Zf2EiOdU87zSRgQYxj/gVex1zPjL/AI97X/eP8qAPM/8AhFl/5+z/AN+//r10HhTSl0ua5fzjJvVRjbjHJ96StDSusv0H9aANbzl9Grq/AEgbVrkAH/Ud/wDeFcdXW/D3/kMXX/XD/wBmFAHoVFFFABRRRQAUUUUAFFFFAHI/ERxHotuWGR539K88Vs4aM59q774mHGg2/wD13/oa8xWUp904oA0NwdvmGDUM8AbkjPvTI7kPw+B71YUk8odw9utAFHZtbGanjlKfeI/CpGj3dv8AGoWTb15+tAFveJF5/OgBkOVNV0cKOOv1p6zgdqALazkryMfSlyrc7ufQ1AsiMOtKVz3oAey+xqMjFDbl6MR9KBMRwQrfUUAKD+IoKRsPmVTRviI5UqfY5oxGc4cj6igCNraIj5GKn36VXkheP74yPUc1dEbHlTke1A47fUHvQBSjmMZ9u9WFmQ9elPNskvMfyMe3Y1UeNomKspU96AJp4YnhLMu8AZxTbGUSFlMO1V4470xZGXqcin/aCiEo2PY0AOvIFkXCoMk4BzzVMJJYwnzAMk8EdKSS9lMe4EjnGQaQ3f2q3ZJ26DKnvmgCa0mM0ZLdQevrU+Mis23uTaxkYySe9W1vY2Yc8HsaALAA9eaMe9IsyNIUBywHSkMyKwUkbj2oAXFLn+9QDkcUpAI7UAMJFJkZ4zTtpIORx9aQJx1NACUmOfT8aXA7mmkDqOaAHHI5wDTGGcZFAGByfwppkVfvMAPzoACvcHmgOwpvnRldwDEe9MNzn7sTY9+KAJ1dT14r2zQv+QFZ/wDXFP5CvDW84puCLjHIDZP5V7hoH/Iv2OevkJ/6CKYC3nSf/rka8VsPHhg1Ga11MGS381whPVRk9D7V7VedJ/8Arka+U9QSS2uZ1gmygkYDecgDJ9OhoYke1WWo297GJtMvUkXOdjH+lWLpLe+i8psRy9Rn/OK8Dg1fUdGmW4iAYg43IeK9C8L/ABKgvv3OsJG3GC6EZ/8Ar0hi6p4XNncyT2qJJGT+8REwSPp2pmlX72ZWPTIFghz82TsBOeTg/wBK7ZLixv499rKJ1A5+bLqPT6VlXmkQXDBoZVSRvu//AFj2+negDbtNQFzbqLlMtjHzjGc+9XI7ZJNrRSFWHGN3Irzi/k1LTUKXkJkEfKS5J2kcfdByB9KZY+MJkdGeQx7T824jH055oA9WN3LDj5i2ODuHBpkmpJ5ZaWNh7pz+hrkbHxvGxP2xkdTyGTqBW9ZatpWqZS3uomYjO3POO3BoAtRTWl/uMM67snhxsI/P+dOMMsIxsJXsU/8ArVXuNNkO5zhlbGWwOn4CoIYJVYiKRoflxlcg4oAui6wuDGzHryKiufIvozHdQRzKB0ZeR+PUVJG1+if8fZZcYy3PFWIoopMC7Cux/wCWgG1hQByWoeCWZTJo0m04yIZm7+xx/P8AOudhOqaReNjMUqgrJE4Iz7ivVGsdi/upSR/tdaZcWsOpW5tr2Lzh2ZThl+lAHFaf4teDbFex/uiOQVB2/lWZ4w0G68UalY3mj63JYWtvgvHEmSeTlhg9cH9K1tc8JXGmMZoY2mt/7+3P/fQ7Vzv2t7KYFdygc4HAP9DQB22sa7psVisLapueJRuUOQ2DxyB+ePevP7bT/Ct9Z3954dvmPiOOCVFZyVeZzwFwx5PGPWuC8VeIIZvHKNfXAmgRWDIXcBTsOAce/wDSuF0nXLvSNZj1GylAnjYsjSAPhucNg9cZzQB6P8GbyYeLNVF/c+Uwtj5izyYG7eoJ579a9nH2dWEbOjGQZG1hhh7fmK+VLzUpr37VcmfNzdSFp24G/nPYADmtKy8Y+IdPgtWkvZJIoWHlecxfCjB2gZ4BwKAPqH7NtXBVcDpuX+tNa3UZLJGDjkg5ryjwd8VLzW/E72uqXEEVpcIFjjchCjeikctn3NXD471GT4kNoq3Nve2W8RBbf5WQnksWPLYA5wcc+1AHo/2dGYBSvttXmoZHnifbcMEQZIbjJH4Uk2qWlmjNPPBEU27t75xnp371BLqulyzul7NakCQRjdjKkjO0570AQ3VzNG5gRpEBP3gpOfyFPEjIw3YLdMngn86zm8Y6PBdRrYTZLITt8zaCAcd+Op/Gob7xjaRySJHaXDyxsiBRGWVi3ofbqaANCSdo5gTbu4Gejj9Kqm8tZZAotpo3LZ3qxP51Sm8RmdIyLKSVHySMBCoHux554qnBr1zFbbYrGOFmGcPOSAc9OBz+dAGoxik2n5w/dTjB/SoZbN7iNttsIznhsYP4+1YWoatfRmW4gu4LQAlmdYjnbj1Yn88VWnTU30wfatUumZVYfaFuACwJz2FAG21rLZy5SUKeMHcCfzBxXRfD3WGuPGVtau0bcSc8Bsgc9OteSfZbeTc9xcNOImR3F3ev857YAYZ+grvvhXbQH4iWVx9htrV9kuxrbJD/ACnOS2ST+NAH0VRRRQAUUUUAY3jD/kR9c/7B0/8A6LavjzNfYfjD/kR9c/7B0/8A6LavjugCSE/v0/3hUFTw/wCvT/eFQUAGM8U77Mf736Ug+8PrVmgBtrak3cPzfxr2966j+z2/56D8q5+1/wCPyH/rov8AOuuNAFEac2R+8H5Vuf2M3/PYflVEdR9a6OgDK/sdh/y2H5Vzj2R8xvnHU9q7jtXJv/rG+poApfYj/fH5VhMcMRjvXUVyz/6xvrQABuelTyD9631NVh1q1J/rG+tADK7X4uybNQ8NDGf+Kftv5vXFmux+MH/IS8Nf9i/bfzegDgfO/wBms7VX3GH6H+lXKoal/wAsvx/pQBRrS0vwvJ4gjkkjuVh8ohSCuc5rNrtvAf8Ax53f++v8qAMr/hWs/wD0EY/+/Z/xr2f4F+FJNG0zV0e6WXzJozkLjHBrmK9M+FX/AB5aj/10T+RoA7H+zj/z0H5Vas7Ervy45x2qWp7b+KgBn2Q/3x+VZ+qWJdY8OOp7VtVS1H7sf1NAGD/Zrf8APQflXF/EZv7Ms7Jm/eb3YccY4r0OvOvi9/yD9N/66v8AyFAHBf20v/PE/wDfVbXhy8F9JcKE2bQp5OfWuOrp/Bf/AB8Xf+4v8zQB1Hk/7VdX4ATZq9zzn9x/7MK5iuq8Bf8AIVuP+uP/ALMKAO8ooooAKKKKACiiigAooooA4z4mHGg2/wD13/oa8uOOteo/Ez/kA2//AF3/AKGvLTlaAEK9xUkU7IeOPxpmaTvQBdS7Vjh+D61JnI7H3rP+pGKekrL90mgCy8atyM59Kg5HOfr2qRZVbr8pp7J5i5br6igCDzSOoqRbpl6frUTx7DzTTgDNAF5LxT/rAQO9TDy5MGNl981lq2PrTw5zkdfrQBobdvXn8KXjtVWO5dfvfMPQ1KJkccZU+hoAmU45FS+YD1/lVYgg56UB/WgCyV7jp7cVIEjnj2Scjse4qssuOR9Kcr5PA/CgCO40+SPlCHX1H+FZ8+5FI28+mK3Yp8Ntb9aLi2jnBwPmHX1FAHIs/wApXjrmkBwOo5rWutNIBKgEetZLqYXwy9OlAD2VgBuAx603KgcZ/GpGY7RuHX9KicFTlTkUASQzbZA/Bx70+4uBNNuC4HTjrVbinL98Y/WgDWWf5RhG9OaUSucfuyahW68tSWGcYGR3qSK73yEFdq9iaAHGSZscKPXmo97gYJwcdqmS4V2KhsletOJBPPP4UAVsuWyXJwMUjYbILNz71ZEYbtikaAdqAINoHfP15pr/AHTjBPtU/ksOmKYyuP4aAGAjb0IPpSAEMCVyPTNLz3FGcUAIFxyRg9sV7loP/IAsv+uCfyFeFu4RSTXufh858PWJ9YE/9BFMBbzpP/1yNfJt7P5V9PcIrMrSNgr6Zr6yvek//XI18t3Gwq9s8Y3o7Byp5BB7g/ShiRjtPbSQl97IGOdrjBBqqYBDdIbTcH78gZ98d6kvLMQzM1soKgZI3459gRxUK3Y+RZk2k8d8k9/pSGaOnateWN550PnW86YAdeN34dMV3Wm+PYbpfL1PajMMGQcfiRXnUjbsqWZkIHAIPHv71JbTR27bYW8vJxtIBOaAPWTJbapYhXkE9qBhJIiQynsQc5H4/nXMa9o94kavFILiMZxJyGB54PfOK5yw1e4srky290yqBwBnn8Mc11um+Ore4ZY9Sh+RuPMROD+FAHL26SA5kPl+gLFs/hn+hqwLrNyIhIpkBz8xIGOOw/nXVXnhmy1SFrzw/PGZOphL4BP49P8APNcTd6dc2NxKLpJVfPKkfd9frQB2WmeKr/TJF8i/mA7xynep9gD0rq7Hx3Z30qJqEBjdxjzYxwPqDXjImkjQublhj+AEAr+HP5VsWl9JDtDkzlgCOQMc9xQB7ZFPBJEz2M8c8ajlY25A+lNS+kV+mQeBgdPrXlthqksU2ArpIBkeWcE11mm+IZbhGi1GFW24AcEbu3+e9AHYw3jbgwdgPerguo5MZba2evb/AOtXPxWqXK+ba3JbI/1b8Y+hzinIl0rFGJjZTnDdcfhQB1Vte/LtdgQTgE9DVHVvCWl61Ez+SI5HGS0fHP071WtL5GUpJt28AEn26+1advMgjxBlkP8AAOGB9jmgDwLx78FpoWkutO8yQZLExnIz/unvXiF7Z3GnXTW90rRspwQykEfga+9C4k/1iiQYwXHUfUVxXjP4VaB4ytXLRiG5x8ssXUGgD5ZW7sJNJSJlZ3QcXHlgMDj+WfxrCnje2lKzI3PPXrXpvi74R+JvD8SR2Vp9rslODLbryfTcOvf6V55qNhqNk5gvrKeHb2kQjH40AUoppIRhHK5II9QfrWv4YEZ8Q2ovVklj8zDpETuKkEEgj09O9ZcFjeXTEW0EkuOuxc1qWOkara3Mci2Vwr/7pWgD2u2TwatilzeX2AqErvkZGKp2Ck9q3dNg8JappMOoWgint3BKyS9R2IOR17V5Zp+hLf3FtNqS+UYwYxHOPlA+o57nsa37rTLC50GXSbZJokXhfKkLJkdDgkEigD0R9B0+W3Cw+agAGNjgZHp06VmS6BGJsLOrMedrkA/pXPT6zeaZ4bjtbS68y62JDGJB5flYGN3zE7u3f0p9x4t1XQ3tbm+t47jT0hAnkjG6RpPUY4xQBp3nhm/C748N6bWz+tUDY38KE3Cz7AcH93n/ABqbRPizpt8Eh1NBY3TyFRGDvG3PDH0ratviF4fuftX+lKiWzBXkYYV88DBHWgDkL63tpYzHeSs28YKSN1HuKqxxwQqFiV3QH+9kH6V6NJd6HfMsRa3nd08xV4JKnuM9qy59F0mZSbaJh3/dsf0oA4/z4THsjgSM4/iArp/hbMf+FhWSybFLRyBVUd9uarXPheJo28iSWN8YHmLn+opPht4c1Sy+MlhfXDRtaJbSp8rH7xXrg0AfSNFFFABRRRQBj+L/APkSNcz0/s+f/wBFtXx/ti/vv/3wP8a+v/GH/Ij65/2Dp/8A0W1fHtAE0KxCZPnfr/cH+NV8Rf33/wC+B/jUsP8Ark/3hVagCVRFuHzv1/uD/Grnlx/32/74/wDr1nr94fWrxoAntY4/tkPzt/rF/g9/rXYeQn99v++f/r1x1p/x+w/9dF/nXamgBgt0yP3jdf7v/wBeui8hf75/75/+vWAOo+tdJQBF5C4++f8Avn/69cg6J5jfO3U/w/8A167M9DXFv/rG+poATZH/AH2/75/+vXLvHH5jfO3X+5/9eunrl3/1jfU0AAjjz99/++P/AK9TSGLzG+Z+v9wf41BSyf61vqaAH5i/vP8A98D/ABrtfi5Ej6h4bLOy/wDEgtui57v71wld38Wf+P3w1/2L9t/N6APPvIj/AOejf98f/XrM1jyYWhBdzkHog/xrVrE8Qffg+h/mKAKfnQf3pP8Avgf416F8NrNL6wvmjlZdsqg7k9vrXmNeq/CH/kG6l/12X+RoA67+xP8Ap4/8c/8Ar16F8NbH7LaXw8zfudf4cdjXJ12/gH/j1vP99f5GgDrNnvU9un3uaiqe2/ioAl2VS1FPlj57mr9UtR+6n1NAGfs9/wBK4b4m6Z9vsrBRN5eyRjyuc8Cu7rk/Hf8Ax62n++38qAPKv+EZP/P2P+/f/wBetPR7VdDaaSSQyiQBcKuMdfertQXX+pH+9QBd/tiD+4/6V2Hw5vo7rWLpUVhiDPP+8K82ru/hV/yHrz/r2/8AZloA9UooooAKKKKACiiigAooooA4z4mHGg25/wCm/wDQ15fnJ9a9O+J//Iv2/wD13H8jXlgODzQA4jHtSBucUoOetKUU/d4oAbx2pc4oEZ7Gl2EdMGgAJwaesrD7vSoir+1GxwOg9+aALizIy4kGKZJACMody1AM9yKUMynj8s0AB46jikGO1SH5vvoc+1I0Xy5Q/nQAit2B4qRSfaq4yDipFbPsRQBZjkZe/FWFKSckbT6iqcb9iOanB2+/vQBMUxz1HqKA2wgA0wS4Oc/hSsQw44b0oAnEwPB5qZTnBHQdCOtZ6v1Geamjc+v5UAXN27743f7S/wBarz2EVwu5QPrUo2kc8H+8KMOvIIYetAGLPpUwfKHfVN7K4U/PG2PYV1AcOPmH40gBRtynI7g0AcqLV2P3SPqKlW0lXHFdJIY2bBVfrUJSE9BigDHWJv4xkVLuUAbMn13CtBo4wMg5qKSNOvT6UAU1ADFgOowcGgEmUsWx2HFSlRnikIzQBD5ksQ2K27vuPNSLe/8APRSMDr60pjU9RTTECOMgUAPW8Hl5YHk9qk89CwUHnGarGD5s8HjFR+Qyg4yD6g5oAul1YZGCKQxg9RiqbRsFCqeM5Oc0qXEnmFmzgDoKAJJrXzFwCa9w0AbfD9iPSBP/AEEV4at7hCX6n7te56Ad3h+yPrAn/oIpgF792f8A65GvjPU7tzrmoTj5UkupHUKc7MseODX2ZedJ/wDrka+J/EEDw+JNQ8qQH/SpPlHykfMfX/8AVQxI1YdSjuo/LnxIp5OcjIp76VZvl7OWSEsBnnIWuYkmkQjcSOMF0PT2YVZg1G6t1BVJHGchlOVIpDNuLRbpG2xTQu3J+bP/ANelbTb6Bs/Z4533cFZT8v0GKhtPE7glbiNvm4AxxW1aavHIMiRj0xk9/SgDnri11VpNq2mCD94uOv58VbjXU4G2yW5VegZpgc+54NdHFfQz/wCtYRsp4H4025hWVeg55B9OaAM621C9tvuybehDRyAYP6Vrpqs+qW4TUreSULx5yMpOPw/H1qg9oyOzbGx1JxnNPtpktFDR/u/MxlRzuoAdJ4ZilR57GWUSjkLMhGT9awi13ZXLCVCZem9BhRjP611sV86qNwK5684FSTPBexGO8hikQjGGUHH40AYmnaq91tjwFuE5DbvlPtmuu0K/t5F8m7jdHA5HBKn1rmW8LWRkBt7iaHHIVmDKPpnkVbtdLvYQFNyrIp+V42KuB6HHUe2aAPSra6W1UISCNuVPOCPb3rRt9VjlYC42uWOVAPKj615/p+o3UDCLUrZmjZuZIsMD7+1bcUUeoP59mfmBzy20++PU+uaANvUJls5fPiZXtn5ZwDkex96s2mrf6RGscqtkZHJH/wCuszTLOOzfzCw8w/eRxtDfn1q4trZ+YZNu5uojcZC9+KAOus5TPHmZ9zE8MowRUxEyEvEwuF9+Gx/I1xa6mbN1UGRCxyBu/wDr1q2fiKKfKtJtP95D/XpQB0STLN8siEMByR2qvPp0UjAy20FzHn51ljBOKbFdxSgPnd2345qyWKYKnIPegDKn8O+HGY/8Sy1Q9SUQLn8qqSeHtJcHyyEzW4/kSA+egKnr2rNvtNhYf6NcqvG7Y7DkUAY114b0vb5UyRScdcfrmuW1bQdMtZD5EmCvbrmuiu9H1FCSMn6njFZF1pV83zSZVf8AZHSgDBNpH5eCN+OOVNVptIt7uNo54jg/88yV/lW81kw/1r84zkjGfzqmtv5jMInJ55JBFAHMTeF7V3cRTtll2lZVVwR0xkjP61Ql8FMfJSKOMwx5JjjJj8w++c5rtXs5Y0HAO09xmqryTxNlkbg9c4xQBwf/AAjmtWVxLcRtdJcMDHHI2JNiHggYII4p8lzrmiixt7e8Pk2rGdmLkM5/ukEdMDGPeu9TWIF5bczZ6dadLqNrfwlBZs3Y+Yox+tAHIW/xK1W0s7u81S2LmV9lpD5eQp75I7V3nw08c2ut+NdM05bcJevDJJcbT8sRC9B9f0rk73wnYXUglCiBs5HlkjFb/wAKvCkOj/EiC9jkeR3ikBLgHqPXrQB9E0UUUAFFFFAGN4w58D65/wBg+f8A9FtXx/sNfYPi/wD5EjXP+wfP/wCi2r5AoAWFD5yfUVV21ch/1yfWqtACKvzD61c3VVHUVPQBYtXxeQn/AKaL/OuuN6n91q462/4+ov8AfX+ddLQBcF4mR8rVv/2nH/cauUHWtmgDS/tOMj7jVx73i+Y3yt1Nb1cq/wDrG+poAtfbE/utXPs2XJ961D0rJP3j9aAFzTpB+9b6mmDrUkn+tb60AMxXonxQsnubnw26soA0G2HP1evPK9Q+In+s8Of9gK2/9moA84/smT/notYniDSZPMg/eL0Pr7V2FYmv/wCsg+hoA5P+yJf+ei16F8NpRpVjfLKN/mSKRt7cVyddR4S/49bj/eH8qAO5/tmL/nm/6V3vw71JJrS9Koww69foa8pr0X4Zf8ed/wD9dE/lQB6J9rX+6as2lwG3YB7Vl1csej/h/WgDQ84ehrP1S6WNY8qTkmrdZetfdi+poAr/AG5P7rVyPj7UUS1s8o332/lXQVx3xB/49LP/AH2/lQBzP9qR/wBxv0p6TC9jZUBXaQeayK0dK/5bfhQBN9kb+8K7f4XQmLXLwkg/6N/7MK5Ouy+Gv/Iau/8Ar3/9mFAHpVFFFABRRRQAUUUUAFFFFAHFfFAZ8PQf9dx/I15T82ev416v8TRnw/bj/pv/AENeWbfagCMk/T+tOD47GglR9+kGOqkj2NAEiy8808PnvUJBpenvQBMGz14oxUWeOacG96AHY9qT60oanZU0AN3U/caZjHSg9aAHHD8MBTfJ/uk00Z9afuoAUI2OT+INPVyp+YE+9M305XoAlyH4HX3NOCZPzHFQ5pwlI/8Ar0AWCBjDcH1pMADgYqDzQfb6Uvn496ALCzbehzUyyg85xj0qiZCe4/KlDsOgzQBohgfqe+KcSRVBLrHDDFWY5VYZVs+2aAJcoT8wGO9MkgVhmI8Hpk0uVb0pjDAPPQGgCB4nHY1GVb0qSzkLQkA7gGOMnNTFlzzQBT2kUhFWGMfY/nUTbaAI9pPSgg07cB3o3A0AMzjtS7wOvWhgp700gCgB29WGMZqNolfqMUufSjPrQBEbYHHJ46V7poP/ACALL/rgn8hXiBbivb9B/wCQBZf9cU/kKYBe9J/+uRr4/wBVihl1y8xtlDXMhYMOR85r7Avfu3H/AFyNfFutTL/bV40cyMwndhu6jDGhiRXuYzbyeZGnmRZ+dVAyB+fNQxCGYmVbaSD5uGjYg47ZFTWWpJNF5VwRuydr7MZ9KsQEJO+xWiG3BOeCe3+cUhkb20zDKSb1HUg4IHvSqkqhhAcgc5PU1M9uzSNvHlooDFw2FXPc/jTZUWKT93dR3KEfLLE/58+1AEsd8wA3kZAyO2ce3rWhDrR34EgHoelY7XewLuRZh1IYDI/E96QvC8m9lxjoFzQB1Vvq7zZDrvA4JV8mp90NxzGu1unK9MVzFu0SYIdlGM5HerQlMkWFuGGTQBpSkIpUA9c4Y54qvvUyO3mtGxHAXn9KrqXCsoYuSMg5xioWcnGRnHUY5oA2Pt6xxpuYMRj8fwq0urxLwhJHDEYrmJg+d4cLxgknOadaSlkxv3nPDYxgUAdpZaykvyxSBT3BPSrjahJGQQvIOc5B/GuDEoLKS3yY5BGMVaj1qKJBDHIMdBg5/nQB258SXCR+UzLJF12OuRTo/EbRx/KnlfTkCvO7rWZA2YWOM84P3qmtNbvjKqpAJEI6lsEUAejjxJcTReW9xNGD3jOM1YhvWzlLsk44EoyM++K5ayupJpD5sQ8sqDuDZxVg3CeXxK6tnGPSgDsLfxHqmnSAmFZYumEGRj2I6fjXSaN400zUGMYkNvLnmOQEc/jXmUV5dIn7ubcPTdjFPl1BLlVS8smZk/5axkqf++hQB7QbqBwNx687iOtc142Nt/wh+pMIjcEw7fKQFmc54GB2zjPtmuM07xDc2v7q3uPMXIHlTtjH/AhxWxNqMlxb7tpgduGR+jf0P4UAdB4U1qDUPCljLBcrcKkCo+BjDAYIwfSrN1q1tGpEsDEdyK81FxcaOphtomiiT7qIThfwPSq03inUGjMZbcPRxQB3F5q+myKR3HQOBWHc6xaBjsABJ/hYGuHvdXd3wyFDnqBwazpNQfcCWweoPSgD0FdejDbW+f8A4F/9amz3ljOn71R9Dk159/aM+TyOPTrT11CbbmUgjPYmgDr55rPbmAAY/Cq8d0rf6tgHHtXMfbgT8jfhnNO+3bjl2KnGMrxQB07XkaHDu5PfjFdH8NbozePLZcfLsfB/4DXmU+oBCGLluMfM3Wuz+EF6tx8RrRFIwIpOB3+WgD6TooooAKKKKAMjxaM+CtbHTOnz/wDotq+R/so/56D8q+uPFn/Il61/14T/APotq+TaAGR22JF/eDr6VS8tf7/6VpJ/rF+tZtAAI0HO/wDSn74v7x/KmHoahoAuW8kYuYzuPDjt71tHVbbP8f8A3zXOw/6+P/eFWD1NAG0NVts/x/8AfNaX9v2n92X/AL5/+vXJirFAHSjX7Qn7sn/fI/xrCe8g8xuW6n+GoF+8Kqv/AKxvqaAL/wBsg9W/75rNMsW4/MevpS1VPU0AWRLFn7x/KrUkSmRvn7+lZY6itV/vn60AM8pf7/6V7N4s8MXWtQ6BcW80SKmjW6YcnJIyfT3rxo19Gzf8gjRP+wZB/wCg0Aebf8IBqH/Pzb/m3+Fcr4u8MXOnXVtHNPESyE/Ln1+le1V538SP+QlZ/wDXI/zoA83/ALIf/nqn5Grthqdv4eR47zfIZTuXy1zjH1qesDxF/roP900AdF/wmmm/887j/vgf416J8M/GmnfYr8iK4P7xf4R6H3rwCvRfhl/x4X3/AF1X+VAHuf8Awmunf88bj/vkf41e03xjYSeZiKfjH8I9/evN60tI/wCW34f1oA9E/wCEtsf+eU//AHyP8ax/EHjTT4I4C8VxyT0Ue3vWLXP+K/8AU2/+838hQBu/8J7pn/PG5/75H+Ncf8QfH2li1s8xXI+dv4B6fWsyuN+In/HpZf77fyoAt/8ACfaT/wA87n/vgf410/grW7XxHPdxWQkQworN5i4zkkdjXiNenfBX/kJar/1xj/8AQjQB6b/Zz/31rrvh5atBrF0zMDmDHH+8Kwa6jwL/AMhS4/64/wDswoA7miiigAooooAKKKKACiiigDi/idz4ft/+u4/ka8pLEdcmvVfih/yL9v8A9d/6GvKCc9e1AEhAYe1A6etNU+gzS7lHYigB3sP1o5B9D9KaxO35evrjpTRI/RuT6k0APOAOfzo7cGm59aWgBwB45p2TTB7UvSgBxY0ZpueaM+1AC0hJpc0GgABpc4pvSkLYPegCUMe5oLVAWxzR5nHv9aAJfMxTldSKhByO9SA4oAlByOKXcRUe7ijNAEpfI+YClyo5RtppgOaOB1FAE63DD7/PuKf5m4ZBz7VVBB5U1IH2/eGPegBFxFKzL/F2xTzceozRhW96jaL0ODQANLn2phkPrSMhXqKbigBd1J5lIeBSZoAfvzRuphpOlAEm6gkmo92KN2elAD817poP/IAsv+uCfyFeDjNe76B/yL9j/wBcE/8AQRTAW86T/wDXI18P6ujnXLzyE3nz5PQ5+Y/5/CvuC9+7cf8AXI18eahpbS6hcSQOwfzGYrwV69xz60MSObQt5io7bCByWJGB7Z/pVlrufaE+0MV52mUcj6+v41NJavbzBJLUSA8sU4AHtyf5U+OOMSYaBwx6h+OM/lSGRXMYvVAugu7oJUHBPYAVAmiXK+XElyrRAllRmOAe/B6Vdazk2k2xIRhgBTnd78cip7YT2iokiM+W4JOSQfrQBSj064U4X51x8oz0NQBNrgEHPTOep+ldG4tLq3/fEW6knLE5x78e9ZN5ZOijy5YjGOBtIyfw6/pQAwvAikYDNwCAP8806K52qQGDkHoR0qlKY443WWaNWUDCgncc+1NhivJk+zRCN+TJvC5bGOmc9KANCK5LFhFIysT1+lWvOmVBnDleTu+bNZcdqwU/aEYbjhJE4FWonkjhPl7ZFB67eQfp3FAF+B4bhQsn7tmOCccD8Kbc2htlIjcPF3ZD938KzDeN5pJUA56YxSG5lwUmD8c5U5AoASSK5yGtyXHQDb2+lZk8rl9jPtKg5QgjBrZN7lRuJJ7BVNXQ0N+gV9ucD/WrnP40Ac4Z7y5mEszMSqhAdxOFAwB+VblnKXIRnA3dO2Pakn0KYt/o0icD5lDjIqgZbrTmaK7TIJyGx0oA6mFrtAdjng5BAz9cUp1BwGSMl/XP9BWXHrsAhXexRcYDf/Xp/wBvjlxsUOo64NAHR2d+hiRlcyc4ZTxWiZvN2m1l2sB91j1NcJJqqQyBHDQtgkHtj8KE8RfZbhRvJUDO7OaAO7W6YQqdURAQOXQHP4HrWrZ3EqQ77a9WWEDJSYAce/auAtfF8e84mOwHkls/pW5YeI7CQ7ZVyf4XT5QR9BQB2f239yPtUMiK3KuBvQj27Y+lUrjTtPvxI4VS6thvs5wynH90/UVHa6xYyRgR3TRN6SDg/jSvdqFZ4wkoJ5ePGaAMK+0YJNgXIAJ+VboGIn6Z61m3eizxRGSaKRVH8ajcv5jiuiOsCR5I94jZepkXcDWCfEOy9cG5hikBBAWXKOPofu//AF6AMV7Z0HDKw7cVEY5EyVIP6VsR6lb3sk0V3GrzsSUMa8IMcZx1zUo0K4urdJdN2zK65Eb/ACsPz60AYL7tqkgHjg4HNVJpXBzG7Z7hhWpNaSQSGO6t3gdeoYEGoZoUdeNoA7YzQBmktLjzmyM8EZ5/wr0L4HGH/hZVqsRBIjl7/wCzXBOEGS3y7ecHPNd18DyrfFC0IAz5Mvf/AGaAPqmiiigAooooAyfFgz4M1r/rwn/9Fmvk/wAs+1fWHir/AJE3Wf8Arwn/APRZr5ToAYkZ3jp1rN8s1rJ98VnUARGM4PSodhq0ehqCgAiUiZP94U4zrk8GiP8A1qfUVAetAE/2hR2NP+3x/wB1qqGo6ANBb6PePlbqKY8o8xuD1NU0/wBYv+8Knf8A1jfU0AP80ehqLyye9FSjpQBEIjkc1qvE29uR1rPHUVrP98/WgCDyW9RX0bLCzaRouMcaZAP/AB2vnivo0/8AIH0b/sGwf+g0AUfs7eorifHWhXOoX1q8LxqFjIO4n1+ld/WB4h/4+If90/zoA82/4RK+/wCesP5n/CuY8V+Hbq2ubcO8R3KSME/4V6xXF+Of+Py0/wCubfzoA87/ALHuP76fnXSeGtctvCdrNHqCSSG4cMvlAHGB3zj1qtWTr33YPq39KAO4/wCFl6R/z73X/fK/41Ytvi5odhu822vTvxjai9vxryaqd91T8f6UAe2f8Lt8Pf8APpf/APftP/iqqah8TNI16FPslvdp5LfN5iqM5H19q8Tra0L/AFc/1X+RoA9D/wCEqsv+eU35D/GsbxFIPEsMEdj+7MLFm83jOfpmsqtHSf8AWSfQUAYv/CKXv/PSH8z/AIV6L8INBubPUdTMjxkNEgG0n1NZFdx8Mv8Aj9v/APrmn8zQB2f9ny/3l/Ouk8F2zwalOWIOYccfUVmVu+Ff+P8Am/65f1FAHU0UUUAFFFFABRRRQAUUUUAcV8Txnw/b/wDXf+hryrAr1b4mnGgW+f8Anv8A0NeWcNx1oAYBSgH0pwj6kEn8aAhzn8s0ANz+dOU7hyPzpdn4Um30BoAcRTD78U85X1prOe65oAbSE46nHvSn7ueOvQ0gyeozQApI7EGlBx1FJgAAY+tOxjp0oAOPX8qUHBpKaeOtADyc03ik78UZNAAUyODikK4604HA5pwZT6UARhsVIpz1oKq3fFNEb/wkMPrQBIfakIphWRf4SR7UvPdWH4UAPDetOD+tV8sOf50hY54oAsgjtxT0z65qksp71Is34GgC3jg7h+VLkg8N+BqBbjHenecD1FAEp9P5GmHHfBppOehpshLLwuaAFIXnBxj1ppXjI5B6GmAtjaR2pw3hecj04zQAlGKlYx/x8e4pgCk/K+fqKAGYpKceOKQ0AJ3r3fQP+Rfsf+uCf+givBzXvGgf8i/Y/wDXBP8A0EUwFvOk/wD1yNfFl/c3EWo3reYJPJncup+YD5sbT3z2r7Tvfu3H/XI18L675LeItRZmIf7TJnjvvNAkX7fW5Xg80tBI46WzJk4/ID075q/Y6ja3Ee5rdbaRW4cAup9tuelcxAwjUmIxg9fugEfjVmSaRtrum9M7jh8knvzSGdxGiyqkgS3Kn7zQthv++TmowNNMimcxmVDyN7ACuKsb65gvvMsz8wOV8zB2+4B6mpV1Rzfs2qRJL5mSxwEIP4dKAOvu49MvRiWSRFK7toYYx36/41UXw7oksiyLdyRHHHyZHB9axm1CzdwYTJGM9C+4diRg8damRogPMt72RdjbhHIzKAe+OSKANW58LxXDCcX3mrgDzHA6Y4qNdBhRcxXsaSL/AHlHPp0Oagt7mSdDLJLHHKucKVBDDthwc5qEXvnbt/lkjg+ZGcDv1IxQBai077MhgjureRsbjH86EZ6HnOKbHpN867hFbzlgQoWQAj3wABmmJIHZllhs2VuVMRKEL246GpILVorE3aTRgLJtHctx0AHJ60ARz6DdXGwtbSR4GNyFTn8qpf2BrAYssbuM42Fe1bMkRjVEuFjSWQbhnbkZ6Ejt+NPit7m3UBFGxgP3icDI9KAMBtK1gdbEkduDVYvdGT5Fe1dRjjoT+PSumkl1eDL2iy4P9xWx9SOlOW61K5QCaMtxkl1yp/PpQBzI8Q3FtGI3DOV/izzn+VTJ4tuG/dzBJUP8MkYIP1relsZP9bPptq6bRuCxAZOfYZJqrNoFhdRqRYpG5HG2Uoc/Rv8ACgCpH4h0wj/TdDs5wf4o02H9KfPq3hX7OjR6MYjn5vLlcFfzY1XPhO1kbbFqHktn/VyMG/lUyeC7aBv9J1RpCOvlx4A/E0AQNe+F3Qv5N8GbIKrJ/iDVB5dEzhYL/HoZUz/6DWhP4asIoNyzT9fu7lP9KqCx0oN+9mnB+qigCt5Gitnybi7h443xhsfkRUZgMLZs9QimX+EHKt9MHirsenaS8mFknIB+bdIo/pT7zQ9NVfMtri6diM7Ttz/9egDN/tjULZ9ssj/Rv6VvaL4qIl2XBwWPytuI/A9qoafFbbWilu9yf8854MFffrWh/ZNvgbCrtjcBwFYHvkelAHRrfx3E2wSb5SBmMjrWfeW9rES89rHAucNuITJz7cmszdcwL5emeXtAyXA5b2A6fzqIaTeuwv8AVZSOfkMsu0Y9M9fyoA6nT00iVWTyYZhHxvg3oyt/v5/pWfq+sSRrHb2EksCoDucS/wDs3Umqi3tslqn7g3ipg+V/q4UAPQL/AFJrJu9Qa6Yy3QjX/nnFGAFQegHpQBuWOvXETB7nUZJewDyFsfgePzp11c2Fy7yRyyW7/e+4Cufp2rmbS3lnmDDCDd1A6Vv2Ol3OsOYo0WOEH5ppBnHrjPGaAIIke6aRUaH5VJDSSYDewyP8K734MaTqFp8ULKe5gCw+TKN6sCM7fY1nQeEtBgRDcyOZM4Cg/e/H/Cu0+G1ta23xBs0s4WVVjkG93JP3egFAHv8ARRRQAUUUUAZXirnwfrH/AF4zf+izXyrsH/PRfyNfVXin/kT9Z/68Z/8A0Wa+VB0oAekYLj94vX0NUPs//TRfyP8AhV6P/WL9aqUARm3+U/vE6eh/wqt5I/56L+Rq433T9Kq0ALHCPNT94v3h2NQm35P7xPyP+FTxf65P94Uw9TQBEbb/AKaJ+v8AhUf2b/pqn5H/AAqwaZQBGlsfMX96nUdj/hVl7M72/eJ19/8ACo0++v1FXW++31oAq/Yz/wA9U/X/AAp/2Q/89E/X/CpafQBALQ5H7xP1/wAK1HtjvP7xOvv/AIVSHUVpt98/WgCD7Mf+ei/r/hX0SVzo+j8j/kHQ/wDoNfPlfQv/ADB9H/7B0P8A6DQBBs9xWRrNhJczRMjKAFIOTWzVW8+8v0oA53+x5/70f5n/AArkfGfh+5mu7Yq8Qwh6k+v0r0iuZ8Vf8fNv/un+dAHnH/CM3n/PSH/vo/4Vnat4QvrhYtksHyk9WPt7V3NQ3P8Aq0+poA85/wCEH1L/AJ623/fZ/wAKpaj4K1BPL3TW/Of4m9vavSqzdX/5Y/j/AEoA85/4Q6+/57W//fR/wrovC/gHVL1bkQzW2VKk5cj19q0q7LwB96++if1oA5r/AIVjrf8Az2s/+/jf4Vc0/wCHGsQO5eW05HZz/hXptSQ9TQB57/wgGq/89bb/AL7P+FdX4E8LX2lXV41y8JEiKBsYnv8AStytPRf9ZL9BQBL/AGfL6r+dbHhy1eC8lZyOY8cH3FQ1o6N/x8P/ALn9aANiiiigAooooAKKKKACiiigDifij/yL1vxn9+O/sa8oLZHB578V6v8AFA48PW/Gf34/ka8oyD1FADsnjA2/jTw3GG5+oqPIHelyeqnNAEysrfdzml37SRiqxGW9KcCVPBoAmElHmAHBx7VH5uR0pQ46cfnQBLlT1ApDtHQVHkZ46elBP93J/GgB529l/SkO0joRTScrgk8+lKCNuM5+tAChVPTdS7V9f0pDtO3ack/pS5HI7DvigBPLUjr+lJ5IP8WKdj3oK56EfnQAzymB65H1pDGw6Dj2p4Vhxn9aUL/tYNAEOGHXP40gcjrU+cd6CAeTg/hQBEJSOjUGd+xqTy4z2P0o2IBwtAEfnvjg0nmuTjdT2VR2FJlR1AoAb5p7gH6imkqeq4PqDUgZewH5Up2MOVGaAIiM/dOfY0DeO1PAA/h49QaeCpoAFcntzT8jGSQPamcDp/8AqpBHuPWgBQ43fKC1PDHGB19qYcDhcn6UjMw6YUelAAxO7mgNim544pQvrQA/ce/NNyN3OcegpypuPJwKmGxRjGaAIQikdSPrXu2gf8i/Zf8AXBP/AEEV4eTkcACvcNB/5AFl/wBcE/kKYC3gJ88AEnyjwK+R9e+Efji817UJrXw9cNFLdSOj7kAZSxIOC1fXp/4/D/u/1qjZ6sLvW9T0/wArZ/Z/lZfd9/em7p2xQI+Q1+DXj0DH/CPXH/fxOP8Ax6nJ8H/H8TfL4cuT9JI//iq+uNN8Q6Zq11Jb2NwXljUOVaNk3KTjcu4DcMgjIrPv/HOj2VvPKjTXPkSrGwigc7iZBGdpxhsNwcZ54oA+XV+EnjwjL+GrgP0BEiD/ANmpj/B3x03B8OXHrnzE/wDiq+rv+Ev0NZp4mvgpt1cuzIwU7PvgNjDEdwMmmN400NEhL3bq0xbEZgk3qFIDFl25UDcuScDkUAfKg+DfjjzAB4fu1T2ePg/99VZi+DXjIrtbR7tGJ+8dnH5PX1ZD4h0y41U6bDcE3PzbQY2CsV+8FbGCRkZwapXHjLS9PurqHUZ1iMM7RKIw0jNtjV2JAXjAfPfjnNAHzMfhD47A40yaRe4Yrz+tSN8KfG8qYk0CTOcfeTp78/0r6UHjvw8YBKt8WVpBGoELlnJUsMDGSCqkgjg4p9v420C7t5Zre+MiRqjHEL5cOSFKjGWyQRxnkUAfLsnwf8cE/LoVzgDAzIjYH4mmp8I/Hyssn9h3YcHgrIqlT6g7q+op/HPh63tEuZtQxEys5YROdgVirFgBlcMCOccimHxvo9vkX9ysT7pOI1eTCI+wuSF4APXPA9aAPmuD4afEaGNkXRrpmkYbmlkQ5Ge53Z61ZXwB8RDsL+HJF2f3PL/+K5FfTUPiTSp9WOmxXW65DFMbG2lgMlQ2MFgOcA5qrqPjDTLDVYdNEnm3T3CQOoBCxlhnlsYyBzjOcUAfN1v4A8fp5hm8PXxJwVKyR8fhvqSb4fePmkQRaFduq8neUQ89uHI/lX0Wvjjw+0DzfbSFXZjML5kDnClBjLAngEZp7eNdBSKKQ3vyyAk4icmIBtpLjGUAPHzY5oA+c/8AhXXjYrhvDl0AW5CSIOPX72KU/Dbxm/L6BdDJ52sg/TdX0drOvvpl7DbQWbXck9vLOgV8Z2FBjoeu/r7U1PGOhOs7fbgFt0LsSjAMoYKSpx843ED5c8ketAHzvH8NfGAIMmjXZ56FEOB/31V//hXfi4RnbpMh543xrkfka98s/FujX/mC2u8tEJDIrxsrL5YUvkEAjAdfzps/i3SY5fJS4LzGPei+WwVjs3hd2MBivOM5oA+dZvht43ZQG8ORTLknCtsP5ZxWfP8ACfxPeg58J31pJ/eWaNlH4bq+jrfx1po0kX+ohraLahJjR5Qm6NH+YheD84+tXH8X6NFetazXEkUqxNMRJbyKCigEsCRgjkdPWgD5Qm+Dvj+3mYwaBcSqepV0GfzPFNj+E/xGBw3hy62+nmRn/wBmr6um8aaJb6dLfTXEyQQMVmJtpMwkKG+cbcrwQcnsaj/4TTSommNzMFVXwojR5GKBFYuyhcqAHGewyOaAPmNfhb49kVfM8Mzlh3d0yfxDVOnwk8ZyxsZNDuojkYjDIR9c7q+nF8Y6E2pfYVvgZtwQnY2wMU8wDfjbkrz16VZ0nxDpmts66bceayKrkFGUlGztcZAypwcEcGgD5ntPhF4uSQPe6bcuAPuIVI/n7VNd/Drxd9rHl+GLqdVHDs6H8Pvcfka+pKKAPkfUPhp4+vF8tPDFwkajj54//iqpxfB7xzjMnh+5BOOjocf+PV9iUUAfJ9l8JvGULENoUoA5Dsy5/IGty2+HvjKELu02YgchQqgD9a+lKKAPniPwF4rEpkOmTMQw4cjp3HBrr/BHhTVdN8Y297d2MkEKoynI+UEg+9dNceJPM1x1utTu9KsHYw28u232vIm7fncGYdOOlbfhy9e+sZ3a8N6iXLxxTlVBdB0PygD8hQBvUUUUhhRRRQBQ1y0l1Dw9qNnb4824tZIk3HA3MpAz+deHf8KX8Vf9OP8A3/P/AMTX0BRQB4AnwZ8Uq4J+w8H/AJ7n/wCJqD/hSfiv1sf+/wCf/ia+hqKAPnk/BLxYQebH/v8An/4mq/8Awozxd/04f+BB/wDia+jqKAPnNPgd4uWRSfsHBB/4+D/8TTf+FGeLvXT/APwIP/xNfR1cZ/wl2pwWFlrN1BaHTLy58lYI93nxqSQjbicMcgZXAxk88UAeRn4F+Lv+of8A+BB/+Jpv/CivF/8A1D//AAIP/wATXscPjxJWG3Sr95JUSSK0WOMS7CrMXOZNuMAHGc/MBjPSAfEmzUSO1hezRKvnM8UaDyIdsZLPl+ceaPu5PXjigDyRfgX4vDAn+z+D/wA/B/8Aiast8EfFhYkGw6/89z/8TXrEvj+2k1OXT7O2f7RDdwwM0jIVZXnWJiNrEgjdnDAdq6+gD53/AOFIeLPWw/7/AJ/+Jpf+FJeLPWw/7/n/AOJr6HooA+eP+FJeLPWw/wC/5/8Aiaut8HPFBYkfYv8Av+f8K96ooA8E/wCFOeKP+nL/AL/n/CvWBol3/Z+nw/Jvt7SOF/m/iUYOK6OigDmv7BvP+mf/AH1UFx4bvpCNvl8Du/8A9ausooA43/hF9R/6Y/8Aff8A9asbXPAesajNE1v9nwikHdJj+lel0UAeQf8ACstf9bX/AL+n/Copvhf4gkVQv2Tgn/lqf8K9kri9N8YXstrNf34Q26FwtvDYSqzMJNiKsrNsYk49OvagDiv+FVeIv+nT/v8AH/CqV/8ACDxNcmPy/sfy5zmY/wCFeoweNrST7Ss9jeWs1qP3sMqpuB3hMcMQeWB69KoyfES0l1Caws7dxPBOqOZSjKyFyhYbWJBBHRsHkUAeY/8AClvFXrY/9/z/APE10PhX4Za/o5uftn2X97t27JSemfb3rsJfiFa2ccrX2n3iJDaRXLTgR7GEhVUH+sJUszYAbHQ84Gagk+JVtc2avo9hc3EoeJbjOzbbB5vKBYh/myQ2Cm4cZ6UAQf8ACG6p/wBMf+/n/wBanx+D9TXOfJ/77/8ArV3YooA4j/hEtS/6Y/8Aff8A9armneHb61eQy+XhgMYaurooAxP7Juf9j86tafZTW0zNJtwVxwa0aKACiiigAooooAKKKKACiiigDlfH2lXmsaPDb6fCZZBLuIzjAxXn3/CB+IP+fL/x8V7HfTi1s5bhl3CKNnI9cDNYMXiuGTSdBvjbsqaxs2gsP3W6Mvz69MUxHnX/AAgfiD/nyz/wMUf8IJ4gH/Ll/wCPivSrXxj4evY53tdWtZFt4/NlYPwqdN2fSrmna1p+r2b3OmXUVzFGSrMh+6wGcHPTgg0AeVf8IL4g72Of+Bij/hBfEH/Pl/4+K9CHjnw/CkIv9UtLWaSNJDG0oOAw459PepX8aeHY7KO7k1e1WCVmRHLfeZfvDHXIzQB5x/wgmv8A/Pj/AOPilHgTXs82J/77FekQeMfD1zBczQatavHaoHmYPwqk4B9xnj60reL9AXT0vm1S2Fs8hiEhb+MDJXHXIFAHm3/CB692syP+Bj/Gl/4QXXx0sz/32K9CPjTRVkeSTULRbMRRSJcecCH8xmUDH1U4PepW8YeH00+G9bVbYW8zlI33feYdRjrx39KAPOD4F1//AJ8v/HxTf+EE8Qf8+X/j4r1a91nTtP01dQvLyGG0bbtmZvlO7pz71Sbxj4fTT475tUt/s8jsiPknLL1GOvHegDzb/hA9f/58f/H1pR4F8QDpZY/4GK9Jn8YeH7WeKGfVrVJJlRowX+8H+6foT3q3qmtadosKy6rdR2yO21C5+8euAKAPLf8AhB/EPezJ/wCBikHgjxCD/wAeGf8AtoK9Ju/F2gWAhN3qtrEJ4xLGS/DIejfQ1aj1rTZVV47yFlac2wIPWUZyn14P5UAeWf8ACEeICMfYSP8Aga/40o8EeIO9l/4+K7+58baOum3V1p91FftalN8UTc4ZwmeeoyetLJ410SzVv7Uv7e0YSyRgM+QdrlM5AwOR07UAefjwPr//AD5f+PilHgfX+9kf+/gr0ZfF2gNqBsf7UtxdKGJiYkHCjLdfQDNUrz4geHraxW7hvUuozcRW7eTyVMjYUkenU/QUAcOPBGvf8+J/77Wg+CNfxxZn/vsV6XbeJNGvNUbTrXUbeW7XIMKtzkdR+Hf0rToA8gPgjxB/z55/4EKafA+vt96wx7hxXsNFAHjZ8B+IO1l/4+KT/hBPEX/Pj/5EFey0UAeOr4G8Q97L/wAfFL/wguv97Lj2cV7DRQB49/wgmvE/8eeB/vinjwProHFif++xXr1FAHj/APwhPiD/AJ8cf8DWmnwL4hPWy/8AHxXsVFAHjv8Awgmv/wDPl/4+KcvgbXx/y5f+PivYKKAPIf8AhB9e/wCfM/8AfYpf+EJ13vZ4/wCBivXa47xXr91b3621nLqNla2nz397awwuI1K5XPmZ49eKAOV/4QvXCMGz59nFeq6PC9vo9tDKNrxxKrD0IArm9MupovE1taL4mfWYZreV5EdYPkKlMHMaKe56118f3T9aAK5/4+z/ALv9awn8P6hH4gvdS0/Vkt477yvNha1Dn5F28NuGMj2rdbm6b/d/rXJa34vure/mt9PtCYLS+tbW5umcfKZGQkBepG1gM+p9qAJfDHgo+HdWN8dQ+1M1r9mbMO1nw24OzbiS3XPbpgDvXl8ANJcXso1JYhcHcoit9o3eYsgZ1DbWIK4yApIJzmqkfxEbVbDzNPtZLcyGCSGVhuBjeZEIbjhsNnHPXrxT7fxPqtva2N/feZdQ3VvJIkFvGCztGrFh684XGKAJbf4cRW1zdOl7GY5jO0YezV2VpSS24sSGAyeMDg8k1FJ8N5ZLGOD+1VQo8jBxbnMO/b/qm37kxt6bipycjgUknxBmEkFwtrCbRY7prkRTb2PlIrAKCAQfmxhgD/XQ/wCEyuhqP9lNpO7VCVKwLcDYYyjPu3kdtpGMdSO3NACWHgh7HxNDrDakZmillcCSHLuJAcqXLdu2ABgdCeasP4Oik1a+vjdtm7aZinl/c8yKOM9+cCPP41Rh+IBntHuo9KZYI7SK4eSW4RQpkYqqc98qT/LJ4ph+ITtYPPa6Q8zQw3M86iYKFSBgGILAE5zxwPfFAFyPwRCmtW+oi9YmBYl2eWMHy4ZY+ue/mk/hVef4fLJawRQ6nJG0Nnb2n+r+WQROz/MAwJDbsEAj60knjy6WdLX+xW+2s7fuvtC7VURebktj04x60yH4ib7NpJ9MNvK0VtPEktwuGjn3bckc5Gw5ABPTGaAMS+8Aarpthcabo07zR6hBLBNKsMaqoeVnwQWyoG9uVzxxjPNdGPAcRgv4/tzZvbWe3J8ofJ5shfPXnGa3PD+sx6/odvqMUTRLNuGx+oKsVP6g1pUAcjp3gSHTvEQ1BbpHiWd7hI2twXDsuD85J45J4APOM4p934Ka61KeVdSaOyuL1b2W38kFjIECEb88LgA9M5711dFAHB3XgbUorew+z6mJ57I28Fu5t1URQxvuJYZ+YnC5xjpwBUw+Hrr9taLV2EupxNDqDm3B8xWdn+QZ+Qjew78Y7121FAGc+jo2tWN+sjL9jgkhWPGch9vf22frXLWnw0itIZIo79VCoFtXW2XfHiVZQWOfm5RRgYBGe/Nd1RQB5/a+CNXmutUnutVFtcT3NwPPS3VllhlihUkLu+Ugx4GSenNWZPh2zahFcjVSREyFRJb7mAEQiKg7uARk4AHJ5zXb0UAcT/wr+eNLWK21kCG3lSZoZrQOkrpEkalhuGcbNwHTJHoKnk8GX882rS3Ou731JPLMq2u2SFAPlRG38KOuABnJ57119FAHCR/Du6hWSKHWIY4JrxbyW3+w/u5GVFUKRvyVG0HBPJ65q5L4IuP7SutQttWEV3eJJHMxtgy7HWNTtG7g5jBByRyeDXX0UAcl/wAIDZ/2e9l9qkED3aXONnOFgEO3P0Gc1P4T8Hp4YZz9pjuG8hLdCtuIzsTOCxySSc89vQCumooAKKKKACiiigAooooArRabYwXRuYbK3jnJJMqxKGOevOM81ZoooAnooopDCiiigAooooAKKKKACiiigArFTwlo0c/mratjzHlWIzOYkd87mWPO1WO5uQM8n1NbVFAHN6z4NtNQgQWTizlUBDIA7EoEKBeGUj5WPfB7g1Yg8G6HBaPbiz3LJB9nkJdsumFBB57+WvT0rcooAxY/COixXKzratvVw65nkIUiQSDALYA3gNgcVtUUUAFFFFABRRRQAUUUUAFFFFABRRRQAVir4T0hIJIBFc+TJuJhN7MUBLbshS+FOeQQBg9K2qKAObg8DaOskjXEcs5aZpAWnkzhmVtrHd8/zIG+bPNTxeDNChuFmjs23qMLmeQhRu3YALYA3c4Hqa3aKAMKPwZocUdxHHayqlyVaVBcy7WZdoVsbsAjYuCORgYpD4L0Jipa0ckEMxM8mZCJPMBc7vnw5LDdnBNb1FAAOKKKKACiiigAooooAKKKKACiiigAooooAKKKKAK97CtxayQPwsqMhx7jFcPp/hPXEttF07ULixNlooYRSxFzJcfu2jTcCMLgNk4JyRXoBAPUUm0eg/KgDz248DamumafDYXdtBPZ6UbJmGRuYvG2QccAhGGeoznFafhTw/qWjjVhqDwuL6YTIVnklZfkCkMzjJ+6Oe+egxXX7R6D8qNo9B+VMRwFp4DuotOmtp5bWRpVsVJ5IxAct279qnt/BdzH4iudQklt2ilu7m4RecqJYI4x26goc+xruNo9B+VG0eg/KgDziTwFqsVnYLY3lvFJa6XFZvtLL5jLKrthtp2ggEbsZBPSsyfQtV8KXMWp3LQTSPfTyqNs9woEsKqQzBSwYFPvY5zjjNetbR6D8qNo9B+VAHlXh/wJqU1jpd5ctBFgWMjwOjIy+TNJIw244yHGB+dXNR8A6odZk1LT7mEu8tx+5+0SQgLLs53IM5BXkdCD1r0naPQflRtHoPyoA5SbwvI3hPSdHieL/QZrZ2JztIicMQM5PbjNZjeENWsvE1zrWmvZTPNJOBBOzKqpII+cgHkGPp3B61320eg/KjaPQflQB51b/Dy5tdAvtPW6t3kuNPt7RJGBGGjZiSeOBluK1fFvhq+1fUdN1HS5lWWyEqGJ53h3BwOQ6cgjHTuK7DaPQflRtHoPyoA8q1DwXq+h+HNSXTRZ3KXOmpDMrByyNHuJ8scls7uMkYPNaNv4Q11HjhhuLKOyXUn1JXYMZdzq3y7cY4Zs+/tXom0eg/KjaPQflQB5hF4A12Tzftd3bO0llHbF3uZZCWSZZNwBXCg4PygDHvni2/hHXxfwANpt1psN3PeC2llkTfK8rOhbCnIUEYHrzXom0eg/KjaPQflQBwJ8NeJZ9T1W8updM8+8ieC2uFd2NrER8qqhTHJ5Jzz+AFZqeAdcNxd3Ms9qZJ3sZEEl1LKQ1vKXIZmXuDxgcenevUNo9B+VG0eg/KgDzzTPB+vW3imz1K+u7eeO2uriRv38h3JJuxtTG1cAgd84zmu9qbaPQflRtHoPyoAhoqbaPQflRtHoPyoAhoqbaPQflRtHoPyoAhoqbaPQflRtHoPyoAhoqbaPQflRtHoPyoAhoqbaPQflRtHoPyoAhoqbaPQflRtHoPyoAhrLvfDek6jcvPe2azSSAByXbDY6ZAODW1tHoPyo2j0H5UAVY7eGJsxQxofVVAqzH938aXaPQflSgY6UhlVv+Ptv93+tZd14W0e91L7fc2pacukhIkYKzp91ioOCR6kVu7VznaM+uKNo9B+VMRz1t4P0O0V1hsyEbbhWlchArb1Cgn5QGAOBxViTw5pUtpBbSWitDbq6RLub5Q4ww655BrZ2j0H5UbR6D8qAOdHg7RPL2SWjTZLlmlmd2femxgxJyQVAGD6Cqep+CLSayWPS/LgnEwlaa5MszNhCmNwdWHBx97HbHNddtHoPyowPQUAc3Y+DtKtNEi054TKFhijeTcysxjOVYYPykEk8VLb+EtFtYriOKzO24jkilDzO25ZMbxkk/e2gn3+tb+0eg/KjaPQflQBjnw9pZvPtZtFM+CN+5uhTZ6/3eKgl8I6JKqbrLaY44o0ZJXVlWPdsAIORje351v7R6D8qNo9B+VAGbpelWejWK2emxeTbqzMqby2CxJPUnuTVypto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgCGipto9B+VG0eg/KgBaKKKQwooooA//2Q==)

**1.2 ALGORITMO DI EQUALIZZAZIONE**

L’algoritmo di equalizzazione si basa su una semplificazione dell’algoritmo dell’istogramma per la

ricalibrazione delle immagini.

È definito nel seguente modo:

*DELTA\_VALUE = MAX\_PIXEL\_VALUE – MIN\_PIXEL\_VALUE*

*SHIFT\_LEVEL = (8 – FLOOR(LOG2(DELTA\_VALUE +1))))*

*TEMP\_PIXEL = (CURRENT\_PIXEL\_VALUE - MIN\_PIXEL\_VALUE) << SHIFT\_LEVEL*

*NEW\_PIXEL\_VALUE = MIN( 255 , TEMP\_PIXEL)*

*MAX\_PIXEL\_VALUE* e *MIN\_PIXEL\_VALUE* sono il valore massimo e minimo tra tutti i pixel che

compongono l’immagine.

Dalla loro differenza calcoliamo il valore di *SHIFT\_LEVEL* che verrà applicato ad ogni pixel.

Si tratta di un numero che va da 0, nel caso in cui la differenza sia esattamente 255, a 8, se invece il valore massimo e minimo coincidono.

Successivamente facciamo la sottrazione tra il valore di ogni pixel e quello minimo ed effettuiamo lo shift\_level.

In ultimo luogo, se il valore calcolato è inferiore a 255, riscriviamo il valore ottenuto, altrimenti 255.

La riscrittura di ogni pixel verrà fatta a partire dal primo pixel successivo a quelli da leggere.

**1.3 MEMORIA**

La memoria è composta da celle di 8 bit ed è strutturata nel seguente modo:

* in posizione 0 vi è contenuto il valore relativo al numero di colonne dell’immagine, inferiore o uguale a 128.
* in posizione 1 vi è contenuto il valore relativo alle righe dell’immagine, inferiore o uguale a 128.
* dalla posizione 3 in poi saranno contenuti i valori di ogni pixel dell’immagine, fino alla posizione (n\_colonne \* n\_righe) + 1.
* dalla posizione (n\_colonne \* n\_righe) + 2 fino a (2 \* n\_colonne \* n\_righe) + 1 saranno presenti le celle di memoria in cui verranno riscritti i pixel equalizzati, con lo stesso ordine dei pixel originali.

|  |  |
| --- | --- |
| 0 | NUMERO COLONNE |
| 1 | NUMERO RIGHE |
| 2 | PIXEL #1 |
| 3 | PIXEL #2 |
| 4 | PIXEL #3 |
| 5 | PIXEL #4 |
| 6 | PIXEL #1 EQUALIZZATO |
| 7 | PIXEL #2 EQUALIZZATO |
| 8 | PIXEL #3 EQUALIZZATO |
| 9 | PIXEL #4 EQUALIZZATO |
| … |  |

Tabella 1 - Esempio memoria 2x2

**1.4 COMPONENTE**

**Immagine che contiene testo

Descrizione generata automaticamente**Il componente avrà la seguente interfaccia:

In particolare:

* **i\_clk** è il segnale di clock in ingresso
* **i\_rst** è il segnale di reset in ingresso
* **i\_start** è il segnale che da inizio alla lettura della memoria e al processo di equalizzazione
* **i\_data (7 downto 0)** è il vettore in ingresso dalla memoria a seguito di una richiesta di lettura/scrittura
* **o\_address (15 downto 0)** è il vettore che manda in uscita l’indirizzo alla memoria
* **o\_done** è il segnale che comunica la fine del processo di equalizzazione
* **o\_en** è il segnale che abilita la lettura e scrittura della memoria
* **o\_we** è il segnale che abilita la scrittura della memoria
* **o\_data (7 downto o)** è il vettore in uscita da scrivere nella memoria

Nel momento in cui **i\_start** = 1 comincia il processo di equalizzazione che al suo termina porterà **o\_done** = 1.

Il componente è costruito per poter permettere successive equalizzazioni, tuttavia sarà necessario

aspettare che **o\_done** torni a 0, per poter permettere a **i\_start** di salire a 1, e quindi ricominciare il

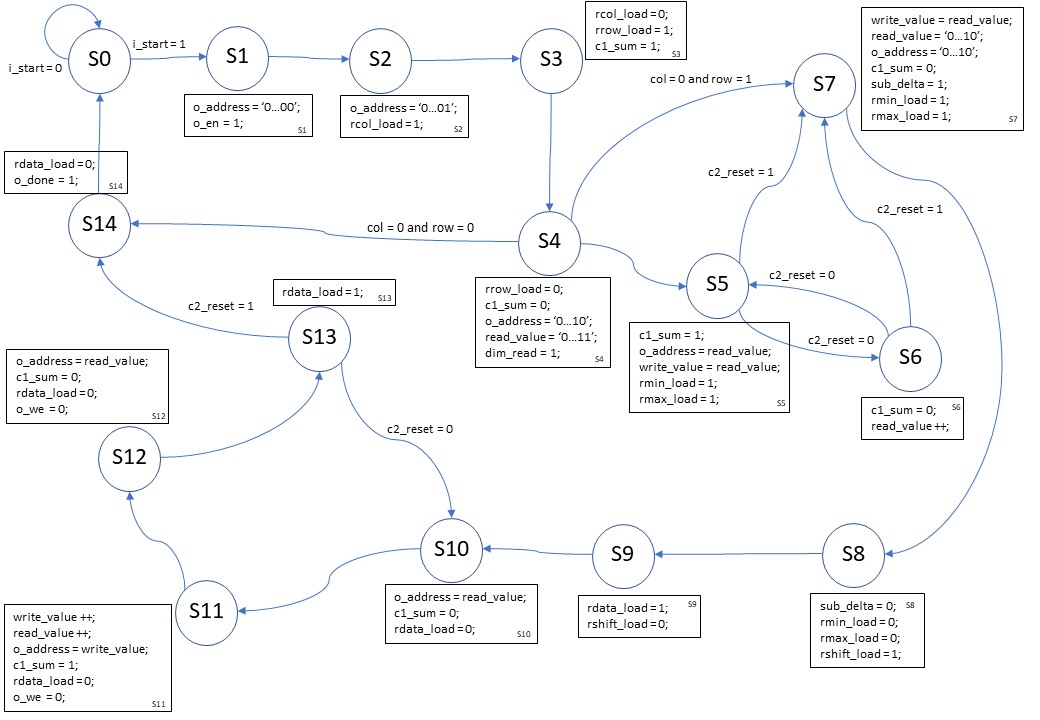
processo.

Si suppone di utilizzare il segnale **i\_rst** solo per l’equalizzazione della prima immagine, pertanto nel caso di equalizzazione successive, il componente dovrà resettarsi autonomamente allo scadere del

procedimento, senza il segnale **i\_rst** = 1.

**2. ARCHITETTURA**

**2.1 FSM**

L’architettura è quella della seguente macchina di stati:

Sostanzialmente, verranno prima letti e salvati nei registri i valori relativi al numero di colonne e righe, e successivamente si procederà a due scansioni della memoria: nella prima si aggiornerà di volta in volta il valore minimo e massimo tra i pixel; nella seconda si ricomincerà dal primo pixel, per poi procedere in parallelo tra lettura del pixel originale dalla memoria e scrittura del pixel equalizzato in memoria.

Per valutare dove si trovi l’ultimo pixel dell’immagine, si fa utilizzo di due contatori: uno per le colonne e uno per le righe.

**2.2 STATI DELLA MACCHINA**

Vediamo nel dettaglio tutti gli stati della macchina:

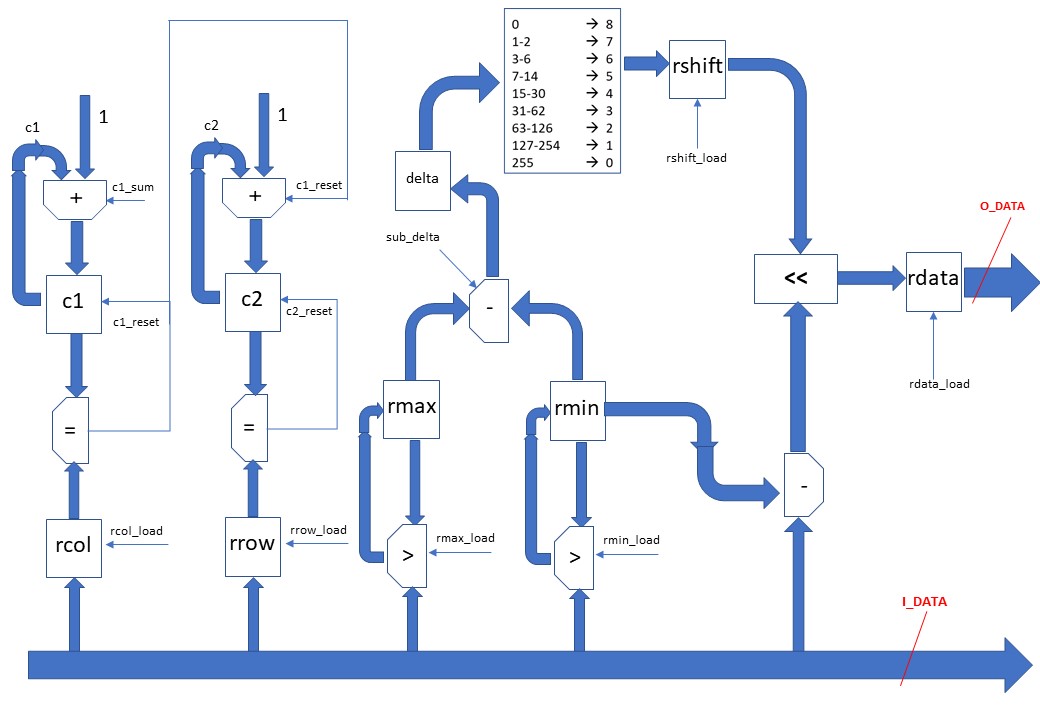
* **S0**: è lo stato iniziale e in cui si torna alla fine del processo di equalizzazione o a seguito di un segnale di reset. Rimane in attesa di **i\_start** = 1.
* **S1**: è lo stato che inizia il processo di lettura dalla memoria, alzando **o\_en** a 1.
* **S2**: salva il valore relativo al numero di colonne nel registro **o\_rcol**.
* **S3**: salva il valore relativo al numero di righe nel registro **o\_rrow**.
* **S4**: inizia la lettura della memoria a partire dal primo pixel. Contestualmente verifica se il

numero di righe o colonne è uguale a 0, e nel caso va in **S14**, o se entrambi sono uguali a 1, e in tal caso va in **S7**, altrimenti va in **S5**.

* **S5**: incrementa di 1 il contatore delle colonne e confronta il pixel letto con quelli minimi e

massimi, eventualmente sostituendone il valore con quello letto nei registri **o\_rmin** e **o\_rmax**. Nel caso in cui si sia arrivati all’ultimo pixel, va in **S7**, altrimenti in **S6**.

* **S6**: sposta l’indirizzo di memoria da cui leggere il prossimo pixel. Torna in **S5**.
* **S7**: nello stato **S7** è appena finita la prima lettura della memoria. Viene salvato il valore dell’indirizzo di memoria in cui si andrà successivamente a scrivere in **write\_value**, mentre viene riportato all’indirizzo 2 (“0000000000000010”) l’indirizzo di memoria da cui leggere.
* **S8**: salva nel registro **delta**, la differenza tra il valore massimo e minimo dei pixel dell’immagine.
* **S9**: salva il valore dello **shift\_value** calcolato che verrà poi applicato nel processo di equalizzazione. Prepara al calcolo del pixel equalizzato con **rdata\_load** = 1.
* **S10**: legge il valore in ingresso dalla memoria, lo sottrae al valore del pixel minimo, precedentemente salvato nel registro **o\_rmin**, e applica la funzione **shift\_level**. Salva il valore del pixel equalizzato nel registro **o\_newdata**.
* **S11**: copia **o\_address** da **write\_value** dove verrà scritto il pixel equalizzato e lo scrive prendendone il valore da quello salvato nel registro **o\_rnewdata**. Incrementa di 1 il contatore delle colonne **c1**, **read\_value** e **write\_value**.
* **S12**: copia **o\_address** da **read\_value**.
* **S13**: prepara al calcolo del pixel equalizzato con **rdata\_load** = 1. Nel caso in cui l’ultimo pixel sia già stato equalizzato va in **S14**, altrimenti torna in **S10**.
* **S14**: stato finale. Pone **o\_done** = 1 e torna in **S0**.

**2.3 DESIGN PROGETTUALE**

**2.4 SEGNALI E REGISTRI**

* **rcol\_load (std\_logic)**: quando a 1, il registro **o\_rcol** salva il dato in ingresso.
* **o\_rcol (std\_logic\_vector (7 downto 0))**: salva il numero di colonne.
* **rrow\_load (std\_logic)**: quando a 1, il registro **o\_rrow** salva il dato in ingresso.
* **o\_rrow (std\_logic\_vector (7 downto 0))**: salva il numero di righe.
* **dim\_read (std\_logic)**: viene posto a 1 dopo che vengono letti e registrati il numero di righe e colonne dell’immagine.
* **rmin\_load (std\_logic)**: quando a 1, il registro **o\_rmin** verifica se il dato in ingresso è minore di quello precedentemente salvato, e nel caso lo sostituisce.
* **o\_rmin (std\_logic\_vector (7 downto 0))**: salva il valore minimo tra i pixel dell’immagine.
* **rmax\_load (std\_logic)**: quando a 1, il registro **o\_rmax** verifica se il dato in ingresso è maggiore di quello precedentemente salvato, e nel caso lo sostituisce.
* **o\_rmax (std\_logic\_vector (7 downto 0))**: salva il valore massimo tra i pixel dell’immagine.
* **sub\_delta (std\_logic)**: quando a 1, salva in **delta** la differenza tra valore massimo e minimo.
* **delta (std\_logic\_vector (7 downto 0))**: salva la differenza tra massimo e minimo.
* **rshift\_load (std\_logic)**: quando a 1, il registro **o\_rshift** salva lo shift\_value calcolato dalla funzione **calc\_shit\_value** con parametro **delta**.
* **o\_rshift (std\_logic\_vector (3 downto 0))**: salva il valore dello shift\_value da applicare per l’equalizzazione.
* **c1 (std\_logic\_vector (7 downto 0))**: contatore del numero di colonne.
* **c2 (std\_logic\_vector (7 downto 0))**: contatore del numero di righe
* **c1\_sum (std\_logic)**: quando a 1, incrementa **c1** di 1.
* **c1\_reset (std\_logic)**: quando a 1, **c1** si resetta a 0 e incrementa **c2** di 1.
* **c2\_reset (std\_logic)**: quando a 1, **c2** si resetta a 0.
* **r\_dataload (std\_logic)**: quando a 1, **o\_rnewdata** salva il valore del pixel equalizzato calcolato dalla funzione **shit\_level** con parametri la differenza tra il dato in ingresso e il valore minore tra i pixel, e lo shift\_value salvato in **o\_rshift.**
* **o\_rnewdata (std\_logic\_vector (7 downto 0))**: salva il valore da scrivere in uscita nella memoria.
* **read\_value (std\_logic\_vector (15 downto 0))**: salva il valore dell’indirizzo di memoria da cui riprendere la lettura dei pixel.
* **write\_value (std\_logic\_vector (15 downto 0))**: salva il valore dell’indirizzo di memoria da cui riprendere la scrittura dei pixel.
* **r\_sum (std\_logic)** : incrementa read\_value di 1.
* **r\_reset(std\_logic)**: setta read\_value a 2 (“0000000000000010”).
* **w\_sum(std\_logic):** incrementa write\_value di 1.
* **w\_read(std\_logic):** copia il valore di read\_value in write\_value.
* **w\_sel:** quando a 1, o\_address assume il valore di write\_value, altrimenti assume il valore di read\_value.

**2.5 CONTATORI**

I contatori **c1** e **c2** permettono la corretta scansione della memoria fino all’ultimo pixel disponibile.

Il funzionamento è semplice: il contatore **c1**, aumenta di 1 ogni qualvolta **c1\_sum** = 1 e successivamente confronta il valore di **c1** con il valore contenuto nel registro **o\_rcol**; se i due valori coincidono, allora **c1\_reset** viene alzato a 1.

**c1\_reset** = 1 comporta l’azzeramento del contatore **c1** e l’incremento di 1 del contatore **c2**.

Allo stesso modo, se **c2** coincide con **o\_rrow**, allora si è arrivati alla fine della memoria e quindi sia **c1\_reset** che **c2\_reset** vengono alzati a 1, azzerando **c1** e **c2**.

**2.6 CALC\_SHIFT\_VALUE**

La funzione **calc\_shift\_value** calcola lo shift\_value che verrà poi applicato durante il processo di

Immagine che contiene testo

Descrizione generata automaticamenteequalizzazione.

Questo valore va da 0 a 8 e viene calcolato a partire da **delta**, la differenza tra il massimo e minimo pixel dell’immagine.

Riassumendo:

* **delta** = 0 → shift\_value = 8 – log(0+1) = 8 – log(1) = 8 – 0 = 8
* **delta** = 1 o 2 → shift\_value = 8 – log(1+1) = 8 – log(2) = 8 – 1 = 7
* **delta** < 7 → shift\_value = 6
* **delta** < 15 → shift\_value = 5
* **delta** < 31 → shift\_value = 4
* **delta** < 63 → shift\_value = 3
* **delta** < 127 → shift\_value = 2
* **delta** < 255 → shift\_value = 1
* **delta** = 255 → shift\_value = 8 – log(255+1) = 8 -log(256) = 8 – 8 = 0

Il valore di ritorno della funzione verrà poi assegnato al registro **o\_rshiftvalue** quando

**r\_shiftload** = 1.

**2.7 SHIFT\_LEVEL**

La funzione **shift\_level** calcola il pixel equalizzato che verrà poi scritto in memoria, a partire dalla

differenza tra il pixel originale e quello minimo e il valore dello shift\_value.

Immagine che contiene testo

Descrizione generata automaticamente

La funzione inoltre limita autonomamente il valore di ritorno a 255, ossia calcola se eventualmente il valore calcolato sia superiore a 255 e nel caso ritorna direttamente 255, evitando un successivo controllo a posteriori.

Lo shift\_level si tratta di uno spostamento a sinistra di tanti bit quanti sono quelli inidicati dallo shift\_value, ossia una moltiplicazione per 1 (shift\_value = 0), 2 (shift\_value = 1), 4 (shift\_value = 2), 8, 16, 32, 64, 128, o 256 (shift\_value = 8).

Per prima cosa se il valore in ingresso è 0, ritorna semplicemente 0, indipendentemente dallo shift\_value.

È facile poi notare due casi limite: se shift\_value = 0, allora non viene effettuata alcuna moltiplicazione, perciò ritorna semplicemente il valore in ingresso; se invece shift\_value = 8 qualsiasi sia il valore in ingresso sarà sempre maggiore di 255: perciò, eccezion fatta se il valore in ingresso è 0, ritorna direttamente 255.

In tutti gli altri casi viene fatto un primo controllo se il valore moltiplicato sarebbe superiore a 255, e nel caso ritorna 255, altrimenti ritorna solo una porzione del valore in ingresso seguita da tanti zeri quando vale shift\_value.

Il valore di ritorno verrà poi assegnato al registro **o\_rnewdata** quando **r\_dataload** = 1, per poi essere scritto in memoria.

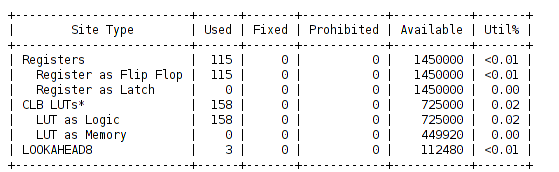
**3. RISULTATI**

**3.1 SINTESI**

Risultati derivanti dalla sintesi:

Immagine che contiene testo

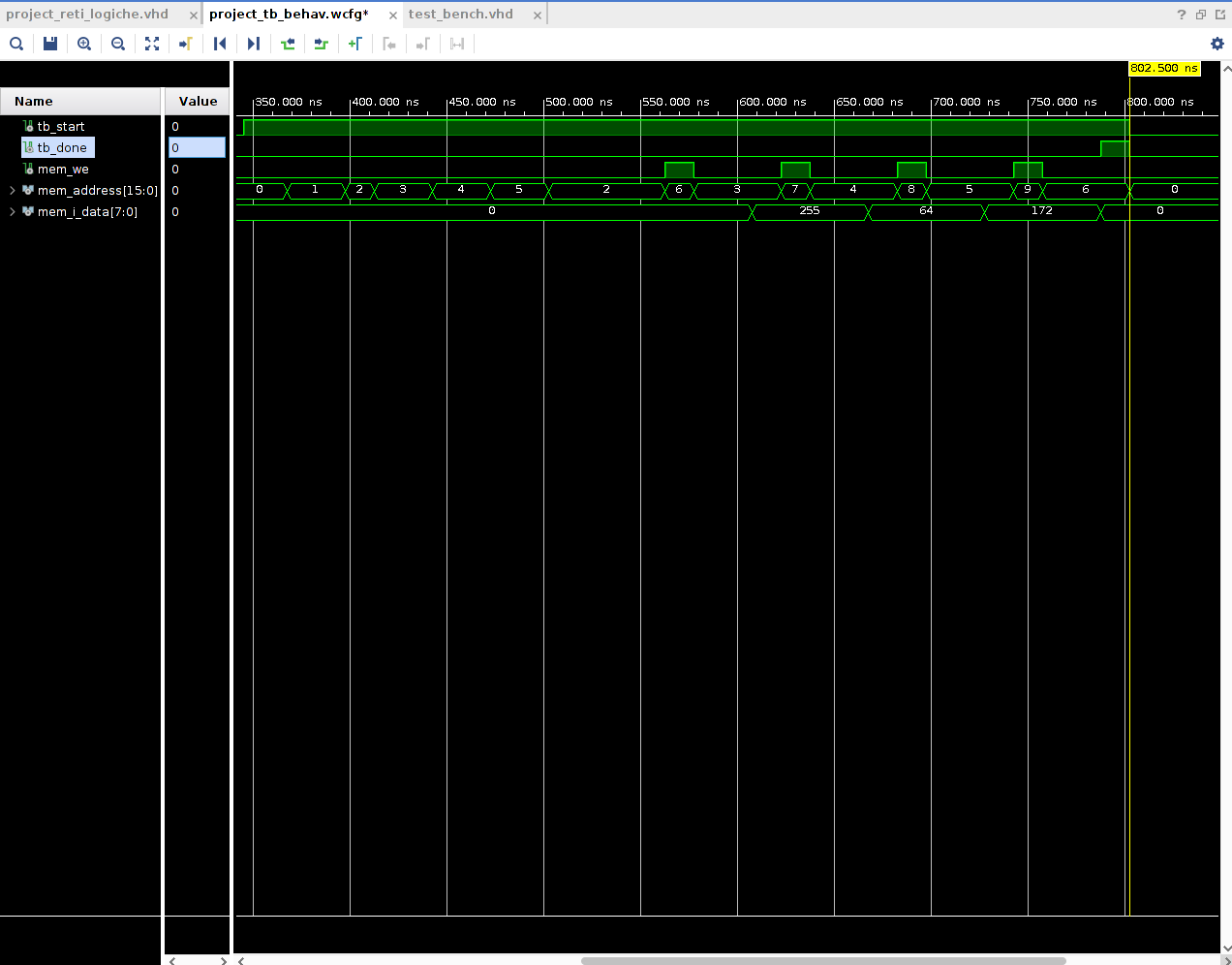
Descrizione generata automaticamente**Immagine che contiene testo

Descrizione generata automaticamente**

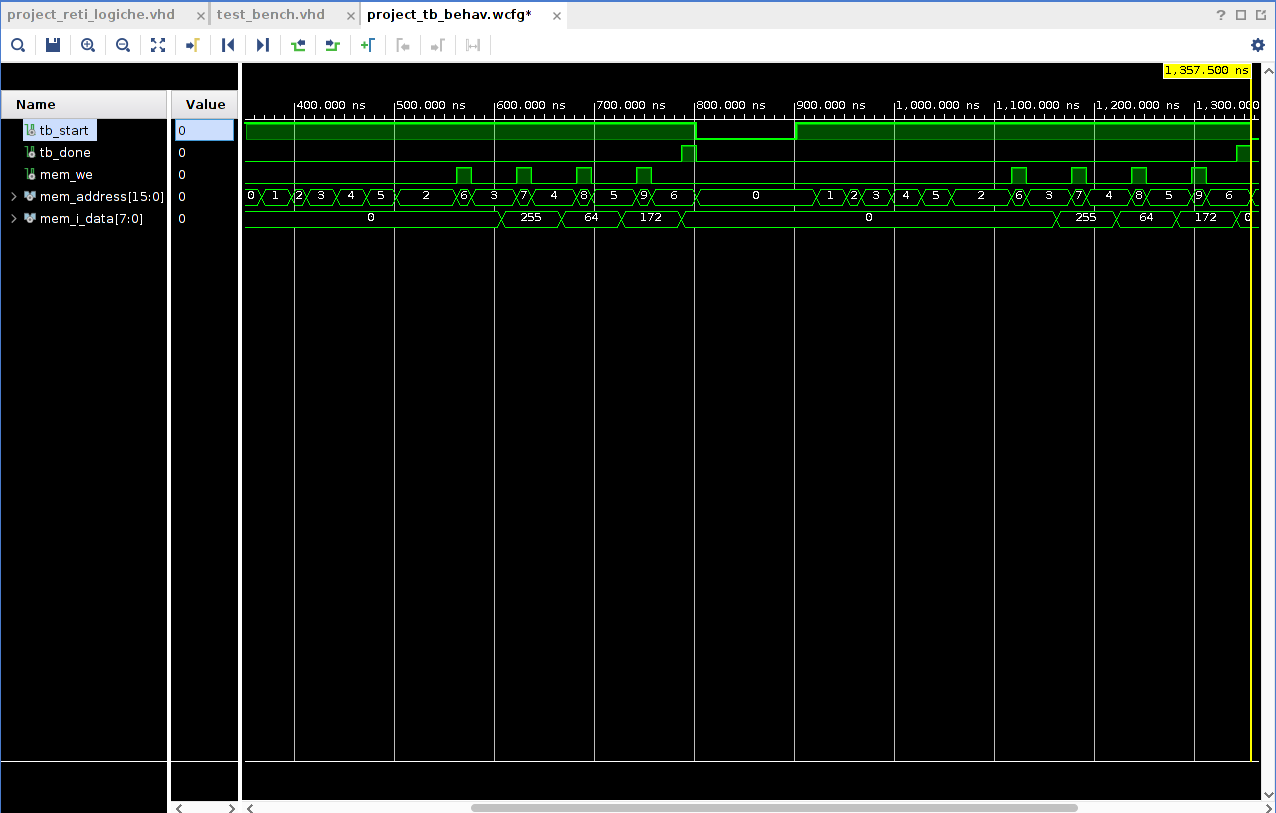
Lo slack generato è di 1.949 ns su un clock di 100 ns.

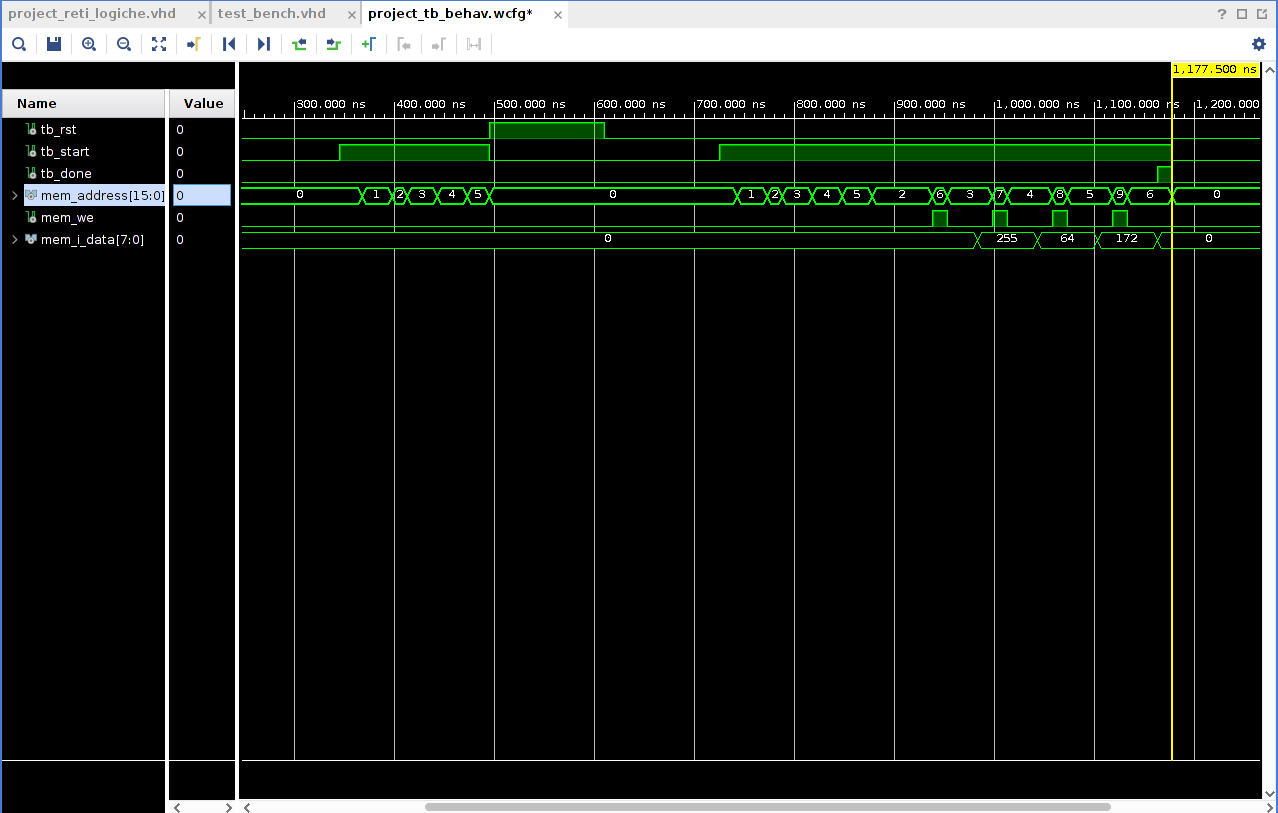
**3.2 SIMULAZIONI**

Oltre al testbench fornito sono stati effettuati altri 17 test:

1. testbench iniziale fornito dal docente.

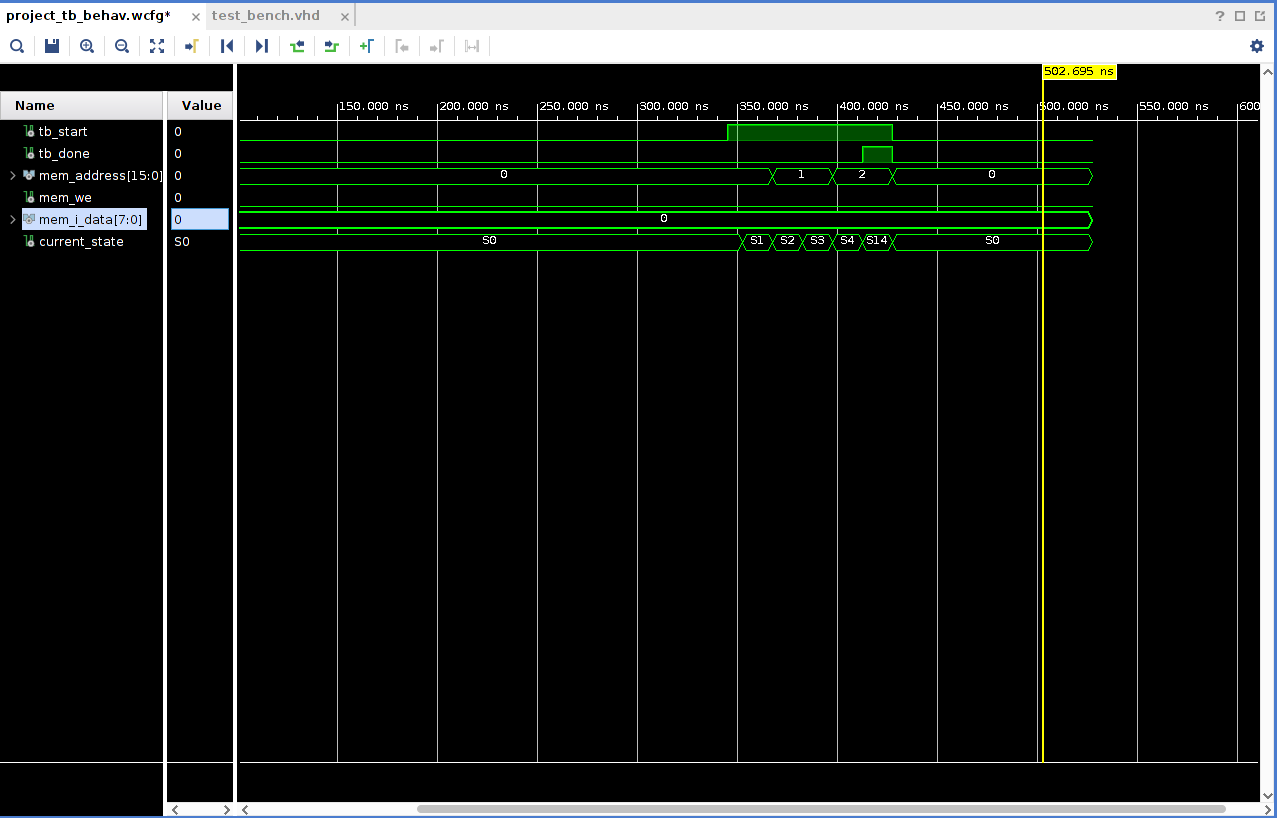
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | **2** | **46** | **131** | **62** | **89** | **0** | **255** | **64** | **172** |

**1bis.** testbench iniziale eseguito di volte di seguito per verificare lo stato alla fine della prima computazione.

 **1reset.** testbench iniziale resettato asincronicamente e ricominciato successivamente.

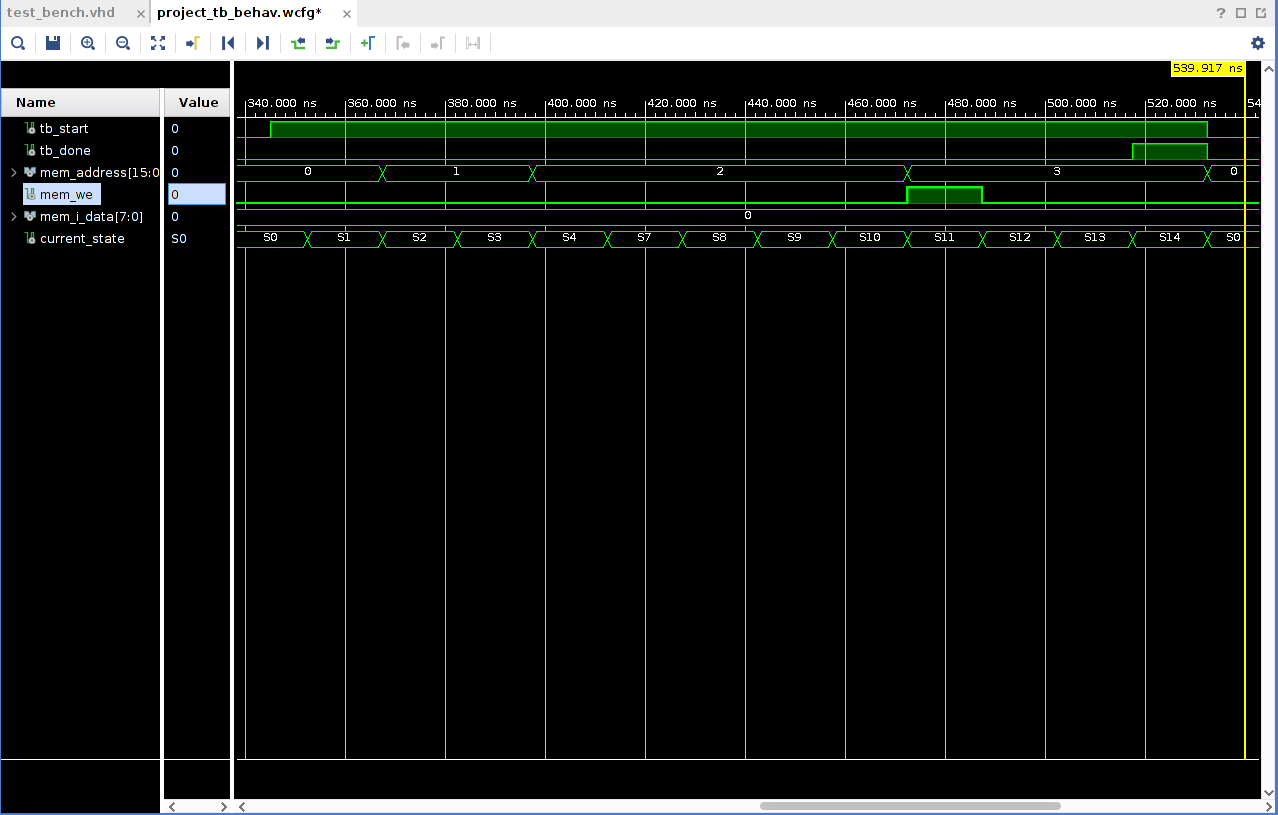
1. test con 0 colonne e 0 righe. Si nota il passaggio da S4 a S14 diretto.

|  |  |
| --- | --- |
| 0 | 1 |
| 0 | **0** |



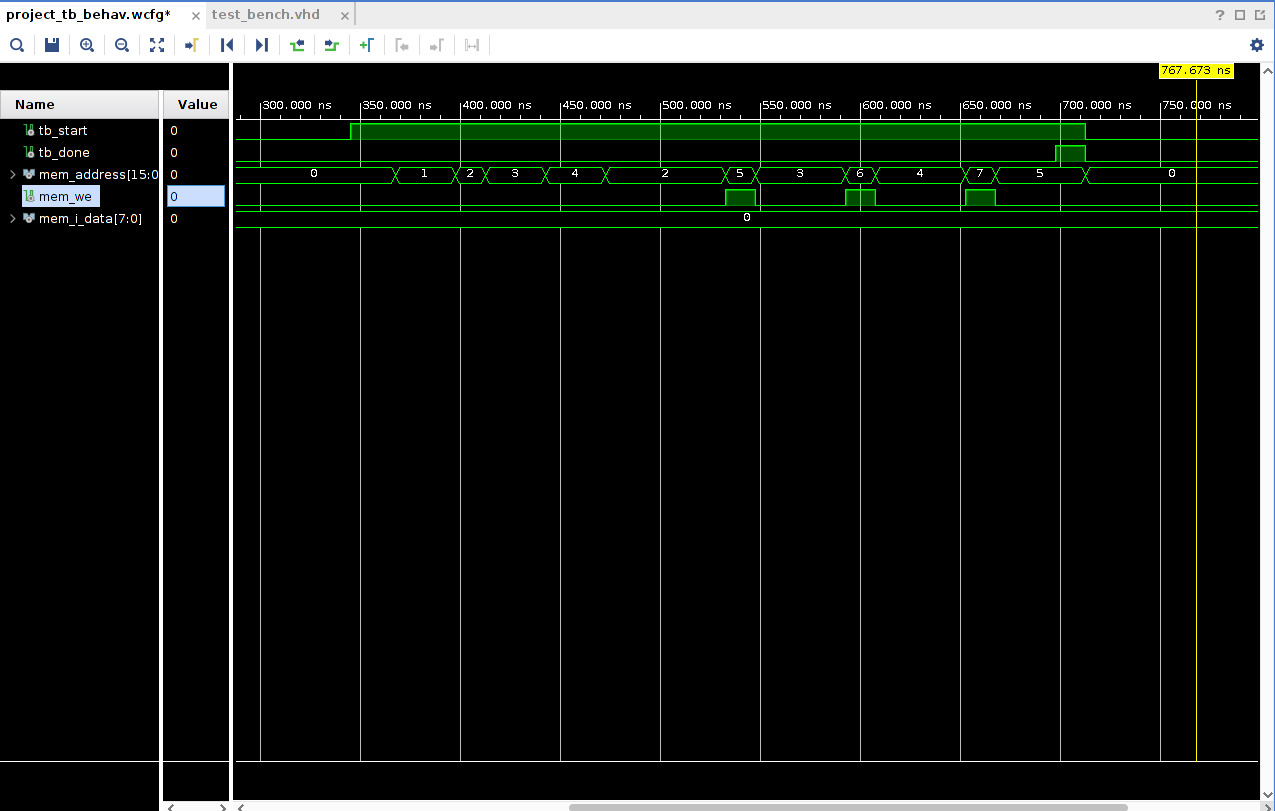
1. test con 1 colonna e 1 riga. Si nota il passaggio da S4 a S7 direttamente.

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 1 | 2 | 3 |
| 1 | **1** | **46** | **0** |



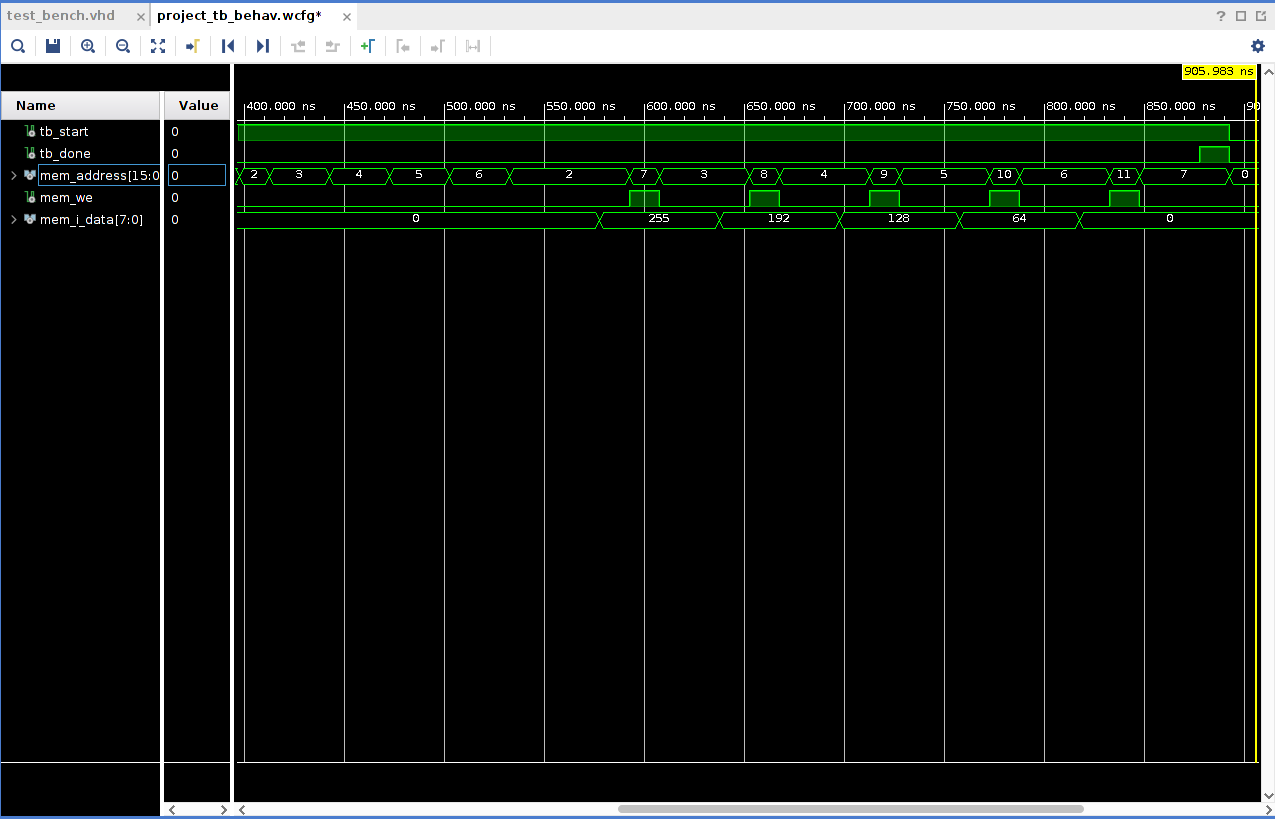
1. test con 1 riga. Inoltre test con valore massimo e minimo coincidenti.

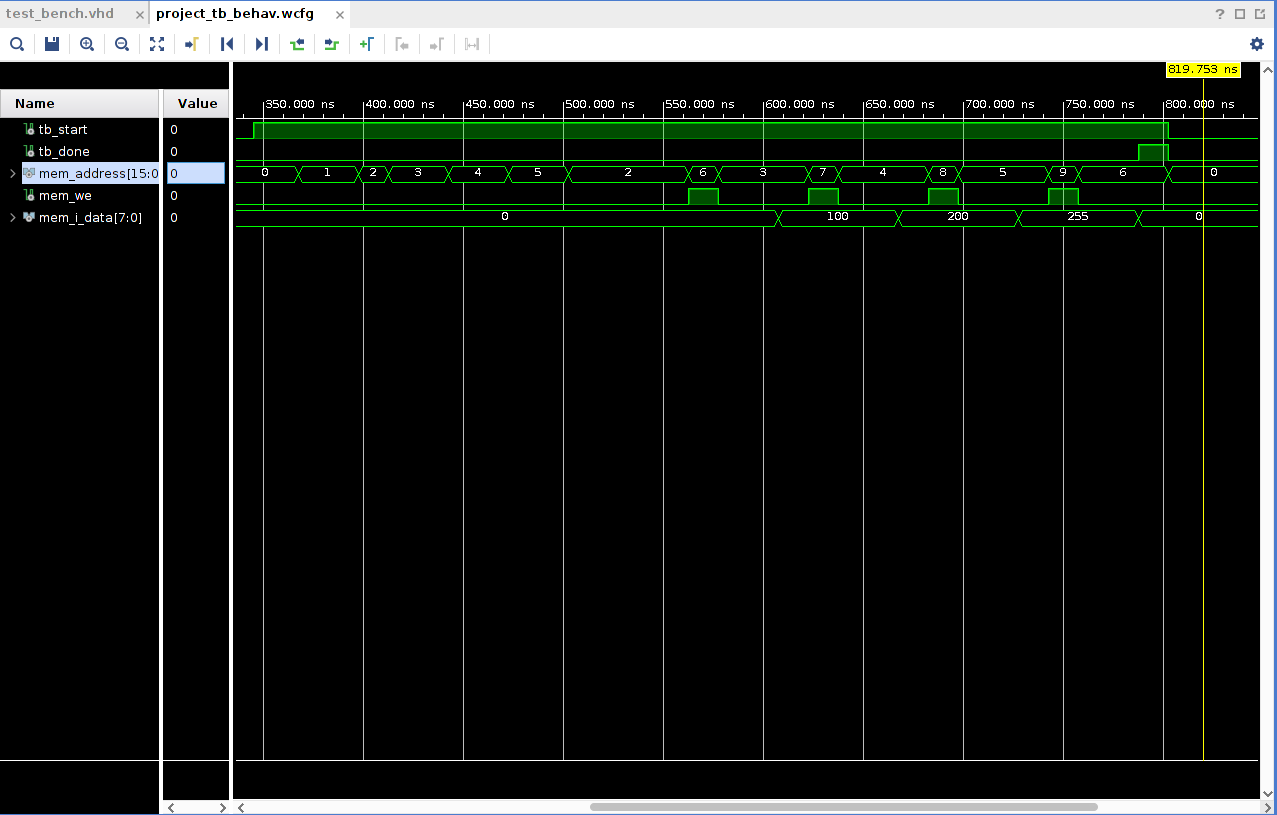
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 3 | **1** | **12** | **12** | **12** | **0** | **0** | **0** |



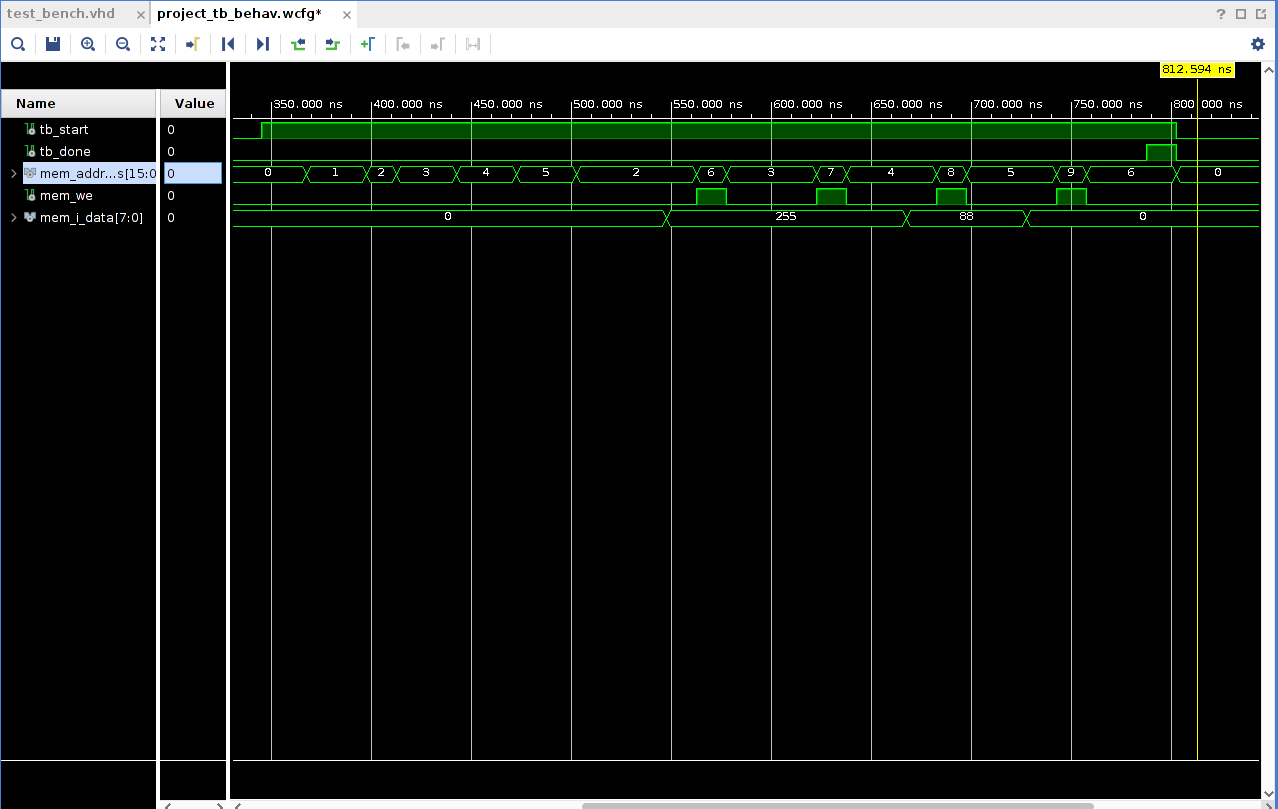
1. test con 1 colonna.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 1 | **5** | **46** | **45** | **44** | **43** | **42** | **255** | **192** | **128** | **64** | **0** |

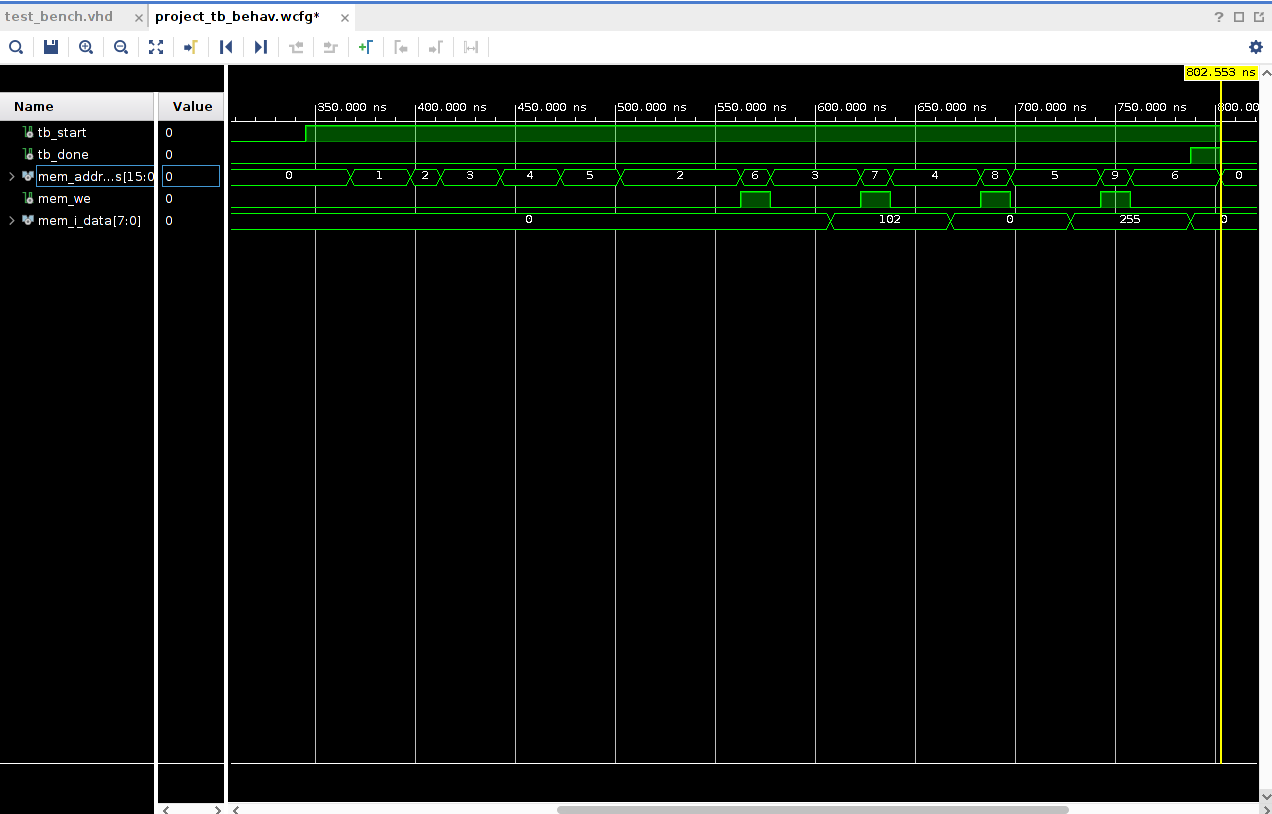


1. test 2x2 con valori incrementali ascendenti.

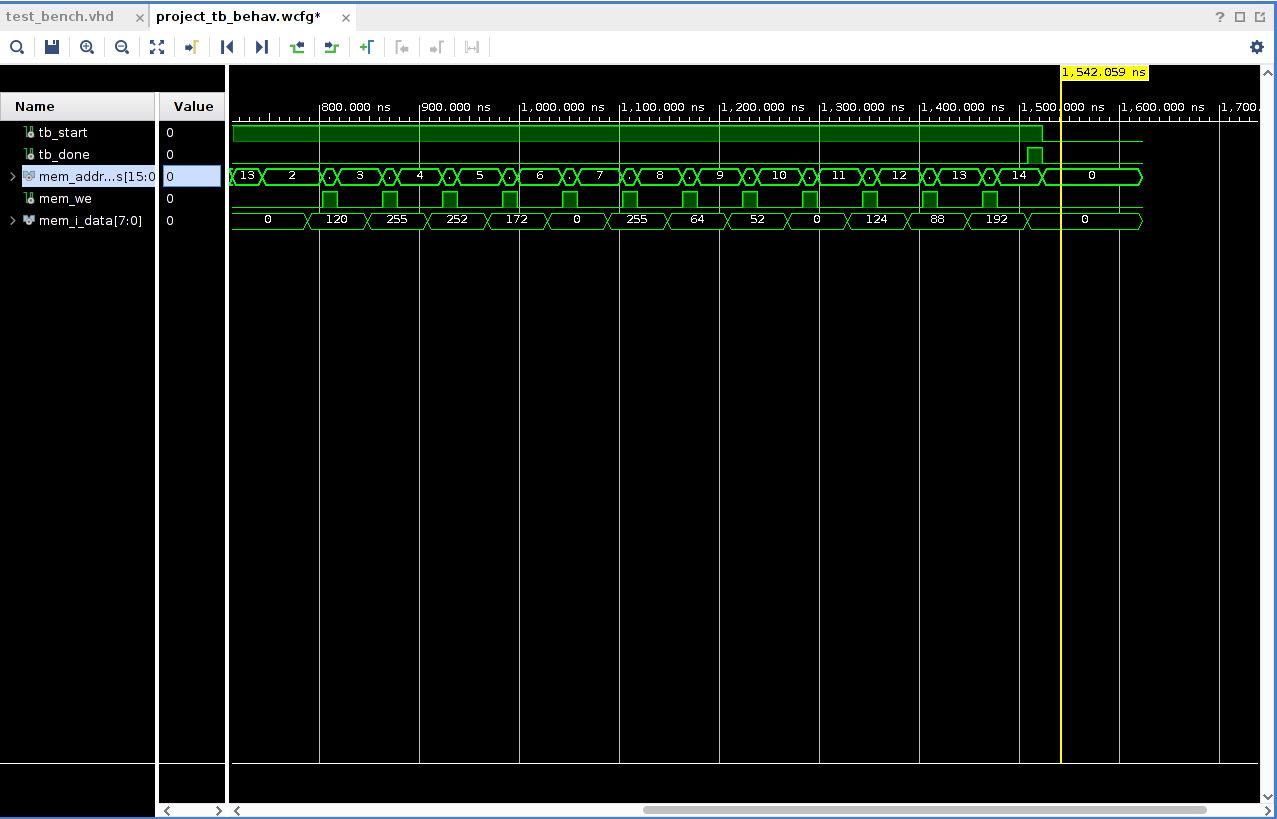
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | **2** | **0** | **100** | **200** | **255** | **0** | **100** | **200** | **255** |

1. test 2x2 con valori incrementali discendenti.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | **2** | **131** | **86** | **34** | **12** | **255** | **255** | **88** | **0** |

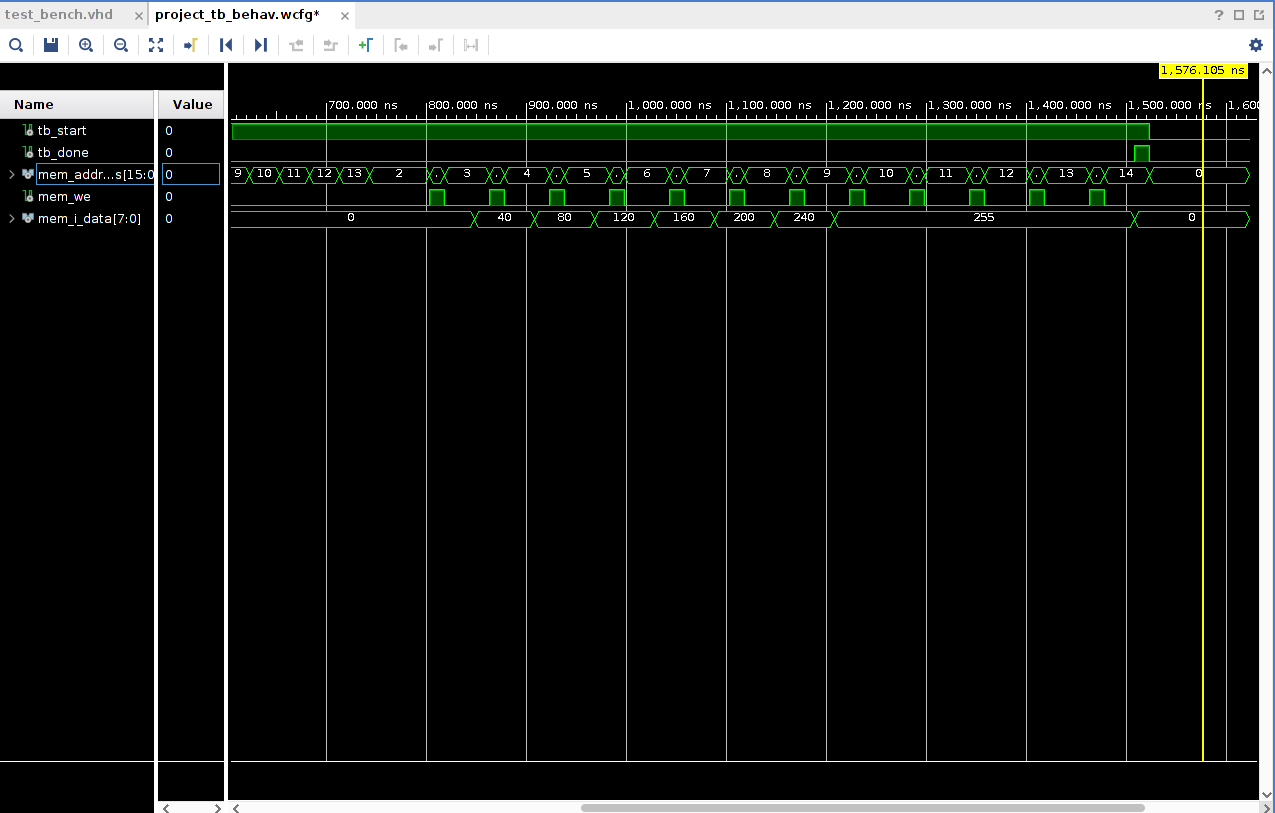
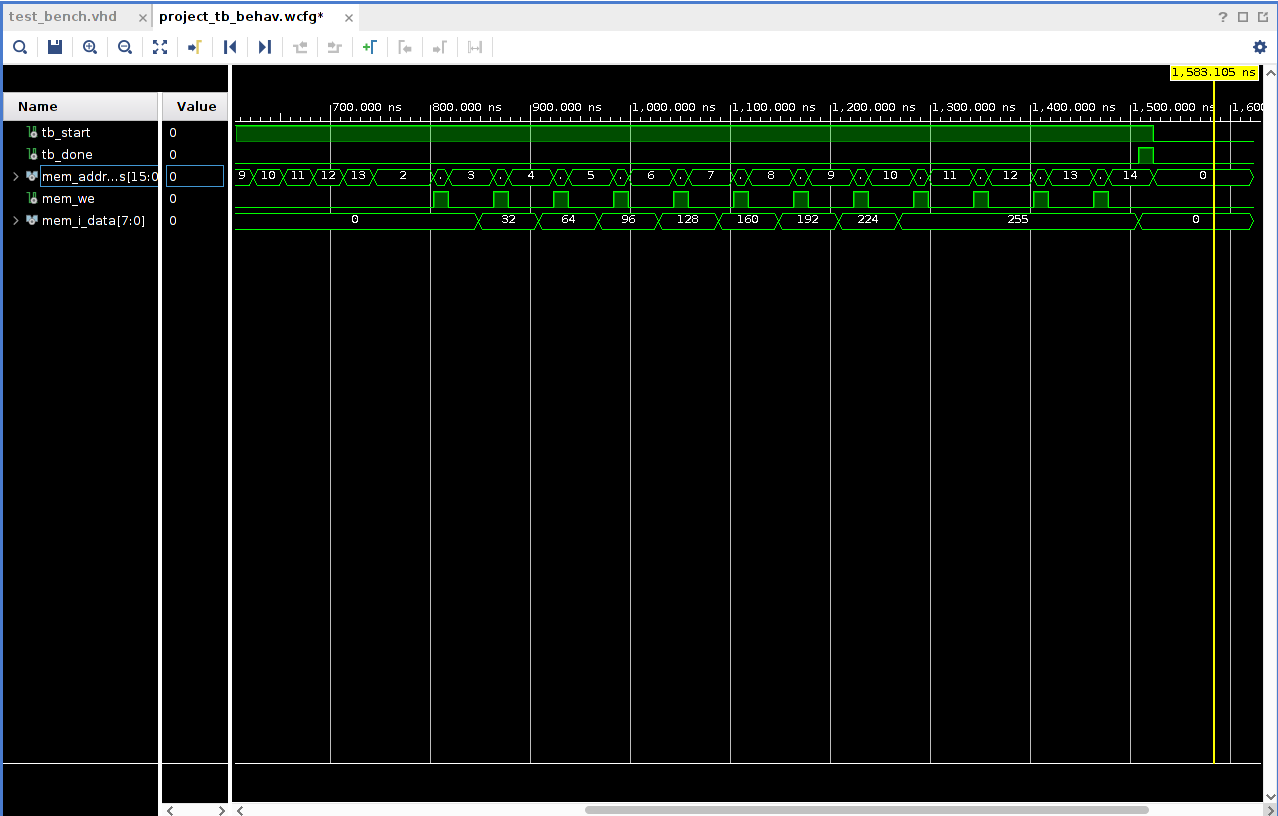
1. test 2x2 con valori casuali

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | **2** | **21** | **72** | **21** | **205** | **0** | **102** | **0** | **255** |

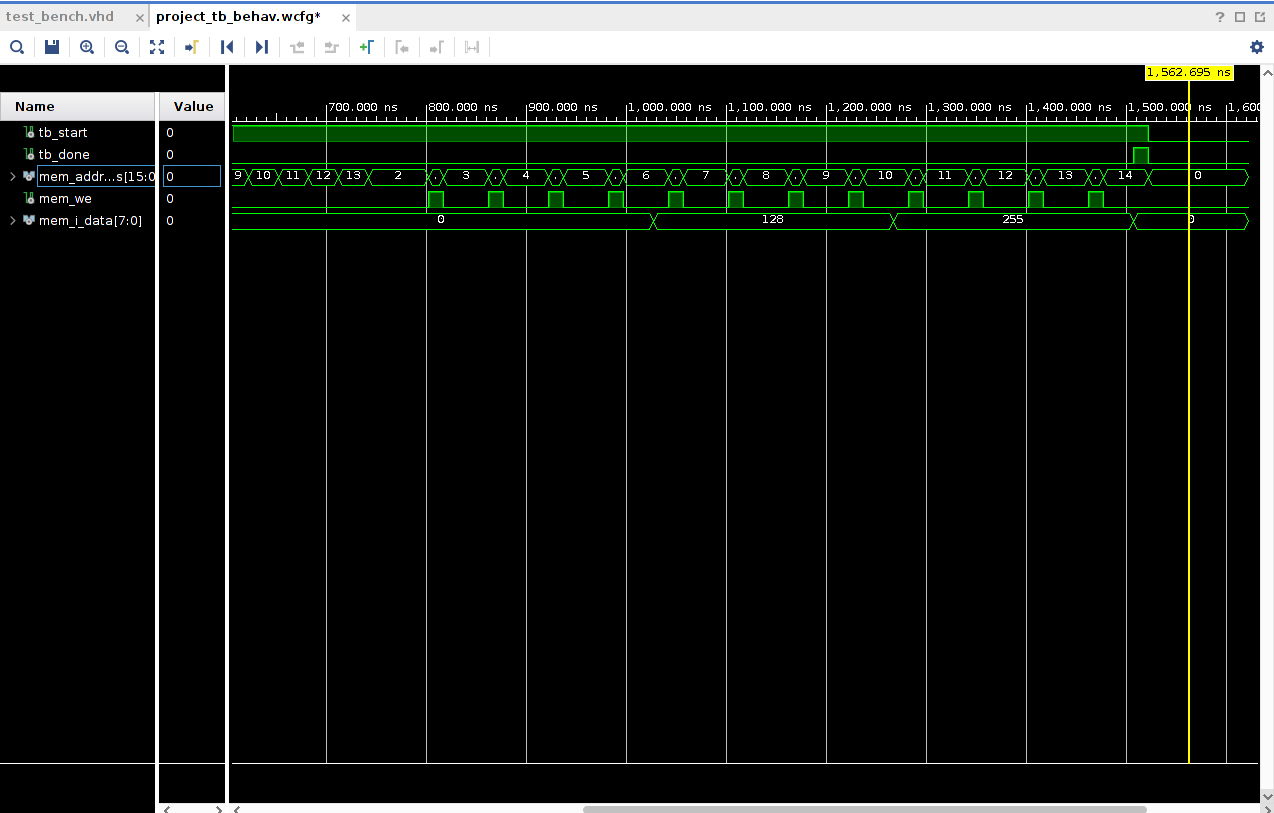
1. esempio 1 specifica 2020/21 fornita dal docente

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 2 | **2** | **76** | **131** | **109** | **46** | **121** | **62** | **59** | **46** | **77** | **68** | **94** | **94** | **120** | **255** | **252** | **172** | **0** | **255** | **64** | **52** | **0** | **124** | **88** | **192** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 2 | **2** | **0** | **10** | **20** | **30** | **40** | **50** | **60** | **70** | **80** | **90** | **100** | **120** | **0** | **40** | **80** | **120** | **160** | **200** | **240** | **255** | **255** | **255** | **255** | **255** |

1. esempio 2 specifica 2020/21 fornita dal docente
2. esempio 3 specifica 2020/21 fornita dal docente

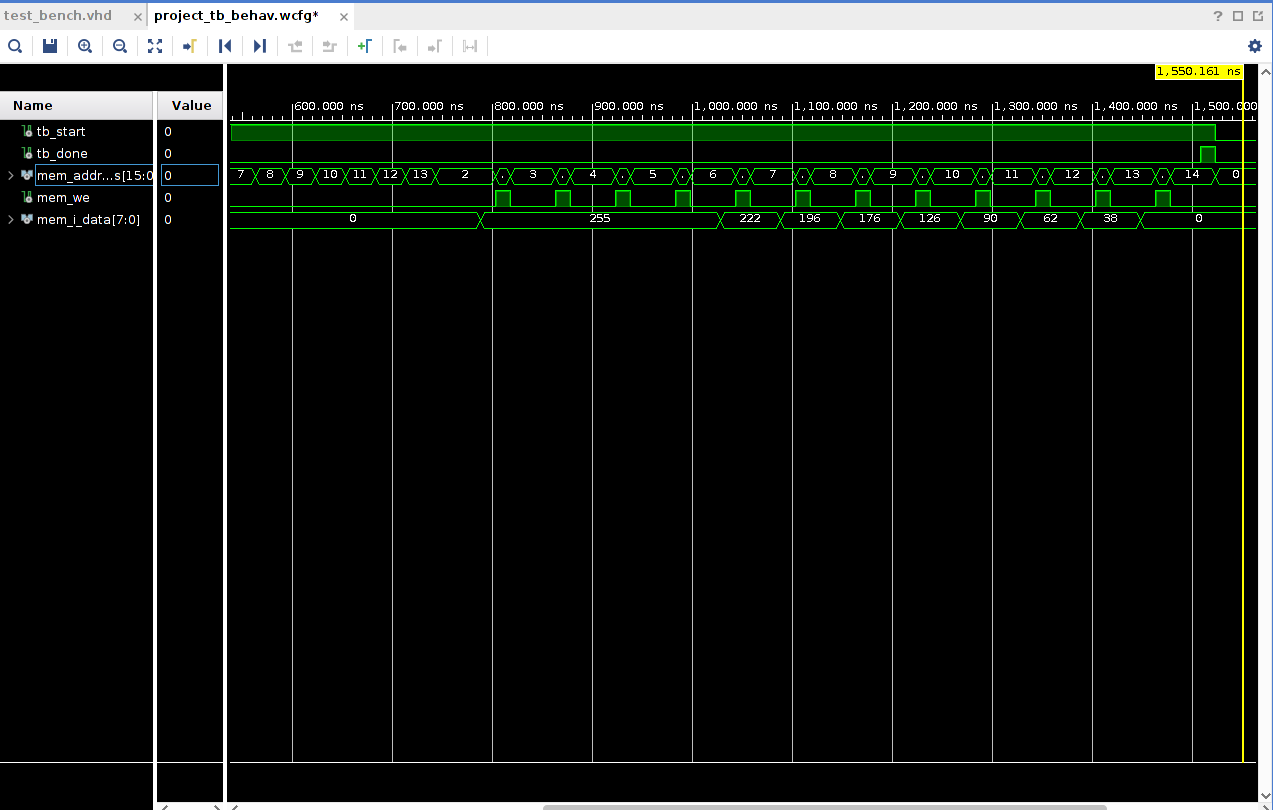
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 2 | **2** | **122** | **123** | **124** | **125** | **126** | **127** | **128** | **129** | **130** | **131** | **132** | **133** | **0** | **32** | **64** | **96** | **128** | **160** | **192** | **224** | **255** | **255** | **255** | **255** |

1. esempio 4 specifica 2020/21 fornita dal docente

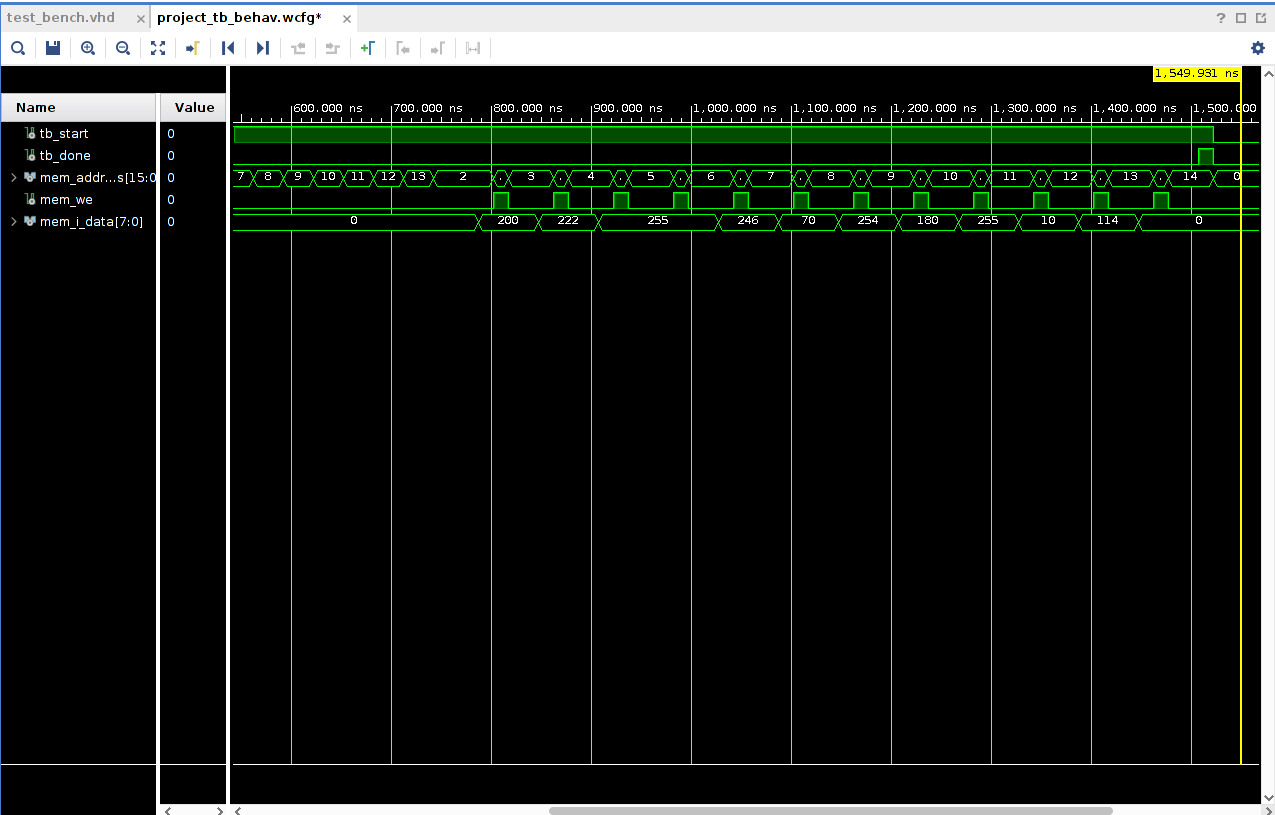
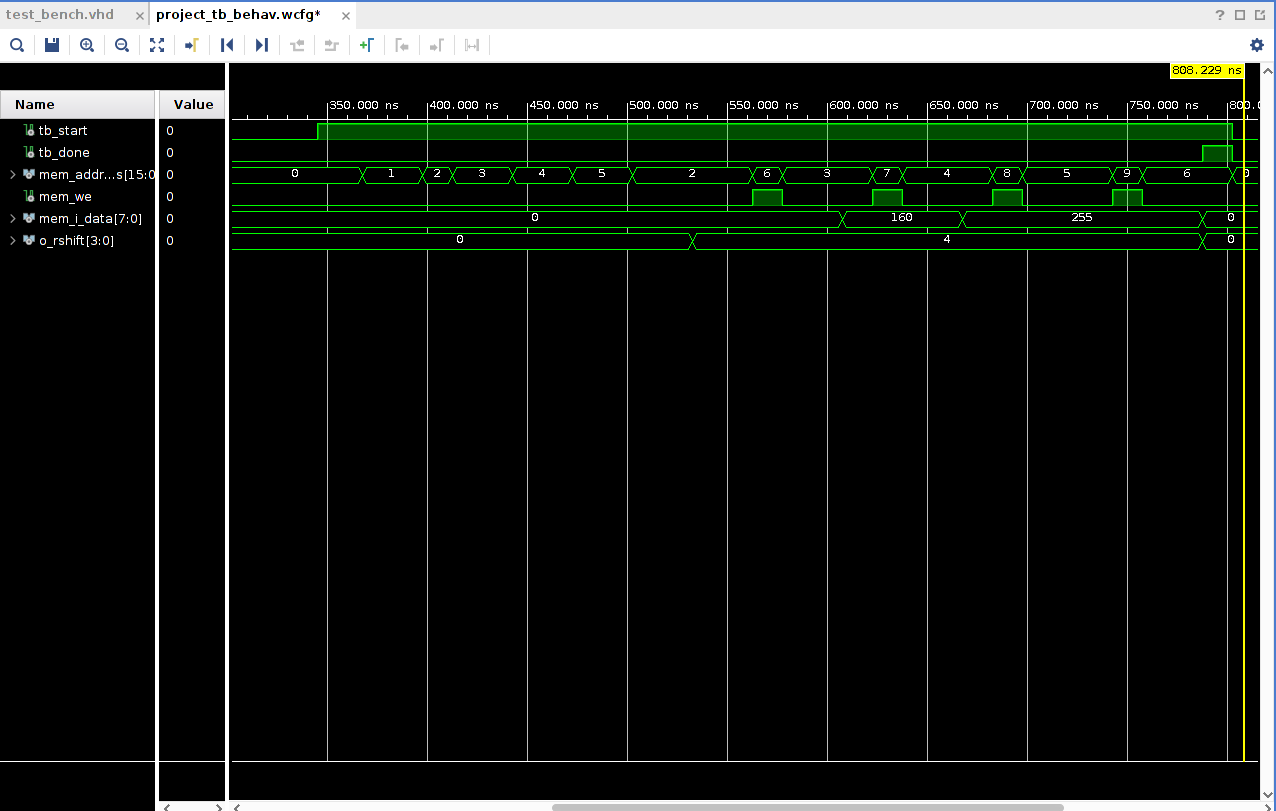
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 2 | **2** | **0** | **0** | **0** | **0** | **128** | **128** | **128** | **128** | **255** | **255** | **255** | **255** | **0** | **0** | **0** | **0** | **128** | **128** | **128** | **128** | **255** | **255** | **255** | **255** |

1. test 4x3 con valori incrementali discendenti

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 2 | **2** | **196** | **182** | **174** | **173** | **112** | **99** | **89** | **64** | **46** | **32** | **30** | **1** | **255** | **255** | **255** | **255** | **222** | **196** | **176** | **126** | **90** | **62** | **38** | **0** |



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 2 | **2** | **116** | **127** | **185** | **214** | **139** | **51** | **143** | **106** | **168** | **21** | **73** | **16** | **200** | **222** | **255** | **255** | **246** | **70** | **254** | **180** | **255** | **10** | **114** | **0** |

1. test 4x3 con valori casuali
2. test 2x2 con shift\_value = 3 (precedentemente non testato)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | **2** | **10** | **20** | **30** | **40** | **0** | **160** | **255** | **255** |

1. Immagine che contiene testo, elettronico, screenshot

   Descrizione generata automaticamentetest 2x2 con shift\_value = 4 (precedentemente non testato)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | **2** | **100** | **110** | **125** | **140** | **0** | **80** | **200** | **255** |

**4. CONCLUSIONI**

Il progetto di reti logiche mi ha permesso di interfacciarmi per la prima volta con la costruzione e

realizzazione vera e propria di una macchina a stati a partire solo da una specifica.

Portando a scelte durante la fase di design progettuale, su ciò che meglio si addice ed è più funzionale, piuttosto che ciò che è più semplice e immediato, ma al contempo poco funzionale.

Ad esempio ho evitato in ogni modo l’utilizzo di moltiplicatori, optando invece per sommatori e

contatori, così come la creazione “fai da te” dello shift\_level.

Inoltre, ho cercato di trovare tutti i possibili punti deboli della macchina attraverso un notevole

quantitativo di test che andassero a toccare tutte le situazioni limite, in cui poteva esserci un

problema.

In conclusione, il progetto mi ha particolarmente stimolato nel ricercare quella che fosse una soluzione ovviamente corretta, ma al contempo senza usufruire di stati della macchina aggiuntivi o scorciatoie, rendendola quindi semplice ma al contempo sintetica.